NOT MEASUREMENT SENSITIVE

MIL-STD-1820
17 December 1992

MILITARY STANDARD

GENERIC TRANSFORMED DATA BASE DESIGN STANDARD

AMSC No. N/A

FSC: 69GP

Distribution Statement A: Approved for public release; distribution is unlimited.

DEPARTMENT OF DEFENSE
WASHINGTON DC 20301-8000

GENERIC TRANSFORMED DATA BASE DESIGN GUIDE

1. This military standard is approved for use by all Departments and Agencies of the Department of Defense.

2. Beneficial comments (recommendations, additions, deletions) and any pertinent data which may be of use in improving this document should be addressed to: ASC/ENES, Wright-Patterson Air Force Base, Ohio 45433-6503, by using the self-addressed Standardization Document Improvement Proposal (DD Form 1426) appearing at the end of this document or by letter.

FOREWORD

The Generic Transformed Data Base (GTDB) standard establishes the file, record, and field formats for digital geographic data bases to be used by real-time simulators.

The GTDB is one of two standard data base formats developed for training simulator use under Air Force Project 2851. The other, Standard Simulator Data Base Interchange Format (SIF), is documented in a separate military standard. The application of one or both of these standards during the acquisition of a training simulator system is expected to reduce the cost associated with data base development for that system, as well as systems which may follow in the future.

The document is structured in four sections, including the main body of the standard and three appendices.

The main standard defines the format of the GTDB as a hierarchy of files, records, and fields. All data bases identified as GTDBs must conform to this format.

Appendix A is a mandatory part of the standard, which gives the detailed definition of each GTDB data field, as an Ada type as well as a range of permissible values.

Appendix B is a mandatory part of the standard, which lists the Feature Descriptor Codes used within a GTDB.

Appendix C is a guidance-only section, which provides illustrative background information on the format defined in the main standard.

For the first-time GTDB user/implementer, it is recommended that Appendix C, "Rationale and Guidance", be studied carefully before reading the detailed design portions of the standard itself.

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

FIGURES

This page intentionally left blank.

1 SCOPE

1.1 **Scope.** This standard describes the detailed design of the data base identified as the Generic Transformed Data Base (GTDB) of the Project 2851 System, and defines the file, record, and field structure of the GTDB.

1.2 **Purpose.** The purpose of this document is to facilitate the production and utilization of Generic Transformed Data Bases.

2 APPLICABLE DOCUMENTS

2.1 **Government Documents.** Not applicable.

2.2 **Non-Government Publications.** The following documents form a part of this document to the extent specified herein. Unless otherwise specified, the issues of the documents which are DoD adopted shall be those listed in the issue of the Department of Defense Index of Specifications and Standards (DoDISS) cited in the solicitation. Unless otherwise specified, the issues of documents not listed in the DoDISS shall be the issues of the documents cited in the solicitation (see 6.2).

AMERICAN NATIONAL STANDARDS INSTITUTE

ANSI X3.27	Information Systems - File Structure and Labeling of Magnetic Tapes for Information Interchange
ANSI X3.4	Code for Information Interchange (ASCII)

(Application for copies should be addressed to American National Standards Institute, 11 West 42nd Street, New York NY 10036.)

DDM-2600-63220-90	National Imagery Transmission Format (NITF), Version 1.1

(Application for copies should be addressed to Defense Intelligence Agency, DIA/DM - 1A, 3100 Clarendon Boulevard, Arlington VA 22201-5317.)

(Non-Government standards and other publications are normally available from the organizations that prepare or distribute the documents. These documents also may be available in or through libraries, or other informational services.)

2.3 **Order of Precedence.** In the event of a conflict between the text of this document and the references cited herein (except for related associated detail specifications, specifications sheets, or MS standards), the text of this document shall take precedence. Nothing in this document, however, supersedes applicable laws and regulations unless a specific exemption has been obtained.

3 DEFINITIONS AND ACRONYMS

3.1 **Acronyms.** For the purpose of this standard, the following acronyms apply.

ANSI	American National Standards Institute
ASCII	American Standard Code Information Interchange
CDBTP	Common Data Base Transformation Program
CIG	Computer Image Generator
CM	Configuration Management
CMP	Configuration Management Program
COTS	Commercial Off-the-Shelf
CSCI	Computer Software Configuration Item
DB	Data Base
DBDD	Data Base Design Document
DBGMP	Data Base Generation/Modification Program
DBMS	Data Base Management System
DFAD	Digital Feature Analysis Data
DLMS	Digital Landmass System
DMA	Defense Mapping Agency
DRLMS	Digital Radar Landmass Simulator
DTED	Digital Terrain Elevation Data
EO	Electro-Optical
EOF	End of File
FAC	Feature Analysis Code
	Feature Attribute Code
FACS	Feature Attribute Coding Standard
FDC	Feature Descriptor Code
FID	Feature Identification Descriptor
FLIR	Forward-Looking Infrared
FP	Formatter Program
FPI	Frames per Inch
GB	Gigabyte(s)
GTDB	Generic Transformed Data Base
HVC	Hue-Value-Chroma
IG	Image Generator
I/O	Input/Output
IR	Infrared
KB	Kilobyte(s)
LOD	Level of Detail
LVT	Local Vertical Tangent
MB	Megabyte(s)
MBR	Minimum Bounding Rectangle
MC&G	Mapping, Charting, and Geodesy
MSL	Mean Sea Level
NOE	Nap of the Earth
QA	Quality Assurance
QC	Quality Control
RGB	Red-Green-Blue
SAR	Synthetic Aperture Radar
SDDD	Software Detailed Design Document
SLOD	Simulator Level of Detail
SMC	Surface Material Category
SNM	Square Nautical Miles
SSDB	Standard Simulator Data Base
S/W	Software

UP	Utility Program
USGS	United States Geological Survey
UTM	Universal Transverse Mercator
VMS	Virtual Memory System
WGS	World Geodetic System
W/S	Workstation

4 GENERAL REQUIREMENTS

4.1 **Media.** Generic Transformed Data Bases shall be distributed on nine-track magnetic tape, recorded at 6250 bits per inch (BPI) in Group Coded Recording (GCR) format.

4.1.2 **Structure/Labeling.** Tape file structure and labeling shall be in accordance with ANSI X3.27.

4.1.3 **Multi-Volume Sets.** GTDBs shall be permitted to span tape volumes.

4.2 **Content.** It shall be possible to generate a valid GTDB which contains only a subset of the total information supported by this format. Those files, records, and fields which may be omitted are identified as "Optional" in Section 5 of this standard.

4.2.1 **Field Population.** Whenever a GTDB includes a particular record, all fields defined for that record shall be populated with either valid or default data.

4.2.1.1 **Valid Data.** Valid data (that is, information traceable back to a real-world source), shall be used to populate GTDB fields whenever such data exists in the Standard Simulator Data Base (SSDB).

4.2.1.2 **Default Data.** Default data (that is, synthetic information which is not traceable back to a real-world source), shall be used to populate fields for which valid data is unavailable. Default values to be used shall be as indicated in the appropriate subparagraphs under Section 5 of this standard.

4.2.2 **Content Specification.** The content of a specific GTDB shall be tailored, as specified by a number of user-provided parameters, specified in the following subparagraphs.

4.2.2.1 **Datum.** The datum of a GTDB shall be one of the following, as specified. If left unspecified, the default datum shall be WGS-84.

4.2.2.1.1 WGS-84

4.2.2.1.2 WGS-72

4.2.2.1.3 International Ellipsoid: Bogota Observatory, Chatham 1971, Chua Astro, Corrego Alegre, European 1950, Geodeti Datum 1949, Hjorsey 1955, Hong Kong 1963, Provisional South American, Qornoq, Rome 1940, or South American 1969

4.2.2.1.4 Clark 1866 Ellipsoid: Bermuda 1957, Cape Canaveral, LC5 Astro, Luzon, North American 1927, or Old Hawaiian

4.2.2.1.5 Clark 1880 Ellipsoid: Adindan, Arc 1950, Arc 1960, Liberia 1964, Mahe 1971, Merchich, Nahrnan, Oman, South East Island, or Viti Levu 1916.

4.2.2.1.6 Any user-defined datum.

4.2.2.2 **Coordinate System.** The coordinate system of the GTDB shall be one of the following, as specified. If left unspecified, the default shall be Geodetic.

4.2.2.2.1 Geodetic

4.2.2.2.2 Geocentric

4.2.2.2.3 Map Projection: UTM, Mercator, Transverse Mercator, Polar, Lambert Conformal Conic, or Local Vertical Tangent.

4.2.2.3 **Terrain Representation.** Terrain elevation data shall be represented in one or more of the following forms, as specified. There shall be no default form.

4.2.2.3.1 Polygonized from a source terrain data base, using the Dirichlet-Delauney tessellation algorithm

4.2.2.3.2 Gridded, with the grid post locations and values extracted directly from the source terrain data base

4.2.2.3.3 Gridded, with the grid post locations and values derived from a Dirichlet-Delauney tessellation of the source terrain data base.

4.2.2.4 **Vertex Normals.** If explicitly specified, GTDB polygons shall include vertex normals; otherwise, these shall be excluded.

4.2.2.5 **Synthetic Culture.** If explicitly specified, a GTDB shall exclude synthetic culture; otherwise; this shall be included.

4.2.2.6 **Boundary SLOD Matching.** If explicitly specified, terrain polygons at different SLODs within a GTDB shall match at the boundaries of adjacent area blocks; otherwise, this shall not be assured. If gridded terrain is specified, or if only one SLOD is specified, this shall not apply.

4.2.2.7 **Convex Polygons.** If explicitly specified, a GTDB shall include only convex culture and model polygons; otherwise, convexity shall not be assured.

4.2.2.8 **Edge Limit.** If explicitly specified, polygons within a GTDB shall be decomposed to meet a specified edge count limit; otherwise, no decomposition shall occur.

4.2.2.9 **Model References.** If explicitly specified, cultural features within a GTDB shall be replaced with model references pointers; otherwise, no replacement shall occur.

4.2.2.10 **Expanded Lineals.** If explicitly specified, all cultural lineal features within a GTDB shall be expanded into areal features; otherwise, no expansion shall occur.

4.2.2.11 **Fragmented Point Light Strings.** If explicitly specified, all cultural point light string features within a GTDB shall be fragmented into individual point light features; otherwise, these shall remain represented as point light strings.

4.2.2.12 **Face Count Exception.** Should a cultural face count limit be exceeded during the transformation process, one of the following shall occur, as explicitly specified. If left unspecified, no GTDB shall be produced in this instance.

4.2.2.12.1 The face count shall exceed the parameter specified

4.2.2.12.2 Features specified in the keep-list shall be deleted

4.2.2.12.3 No GTDB shall be produced

4.2.2.13 **SLOD Parameters.** The following characteristics shall be as specified for each Simulator Level of Detail (SLOD) within the GTDB.

4.2.2.13.1 **Keep-List.** If a keep-list is specified, the GTDB shall include all features listed therein, subject to the exception noted in paragraph 4.2.2.13.2.

4.2.2.13.2 **Delete-List.** If a delete-list is specified, the GTDB shall include none of the features listed therein.

4.2.2.13.3 **Level-List.** If a level-list is specified, terrain polygons in the GTDB shall be flat beneath those features listed therein, subject to the constraints implied in paragraph 4.2.2.15.5.

4.2.2.13.4 **Thinning Tolerance.** If a thinning tolerance is explicitly specified for areal and lineal culture features, vertices in each such feature shall be deleted to the extent that the resultant feature reflects the original feature within the specified error tolerance.

4.2.2.13.5 **Highest Level of Detail.** The culture data in each SLOD shall include features up to and including the highest level of detail specified for that SLOD.

4.2.2.14 **Area Block Parameters.** The following characteristics shall be as specified for each Area Block (AB) within a SLOD. These values shall vary from AB to AB, as specified.

4.2.2.14.1 **Area Block Dimensions.** ABs shall be of dimensions within the range of 0.001 arc-second to 15 arc-minutes, as specified. There shall be no default AB size.

4.2.2.14.2 **Face Count Limit.** If explicitly specified, the number of polygonal faces used to represent culture data within an area block shall fall within the specified limit; otherwise, there shall be no limit.

4.2.2.14.3 **Model Reference Limit.** If explicitly specified, the number of model references within an AB shall fall within the specified limit; otherwise, there shall be no limit.

4.2.2.14.4 **Terrain Goodness-of-Fit.** If explicitly specified, terrain polygons within an AB shall conform to the specified goodness-of-fit tolerance, subject to the constraints implied by paragraph 4.2.2.15.5; otherwise, there shall be no goodness-of-fit imposed. This shall not apply to GTDBs for which only gridded terrain is specified.

4.2.2.14.5 **Number of Terrain Polygons.** If explicitly specified, the number of terrain polygons generated in an AB shall meet the minimum and/or maximum counts; otherwise, there shall be no minimum or maximum.

4.2.2.15 **Additional Islands.** If explicitly specified, additional SLOD Islands shall be included within the GTDB, in accordance with the user's dimensions.

4.3 **Data Formats.** The following formats shall be used to represent information within a GTDB.

4.3.1 **Non-Texture Data.** All data except texture shall be represented in the eight-bit American Standard Code for Information Interchange (ASCII), in accordance with ANSI X3.4.

4.3.1.1 **Field Format.** Data values within all fields shall be right-justified, and the leading bytes filled with the ASCII SPACE (hexadecimal '20') character.

4.3.1.2 **Inter-Field Separation.** Adjacent fields shall be separated by the ASCII NULL (hexadecimal '00') character.

4.3.2 **Texture Data.** Texture data shall be represented in the National Imagery Transmission Format (NITF), Version 1.1, as amended by Section 5 of this standard.

5 DETAILED REQUIREMENTS

5.1 **Data Base Structure.** This section describes the format in which the Generic Transformed Data Base products of the Project 2851 Data Base Facility shall be produced.

5.1.1 The general architecture of the GTDB shall be as illustrated in Figure 1.

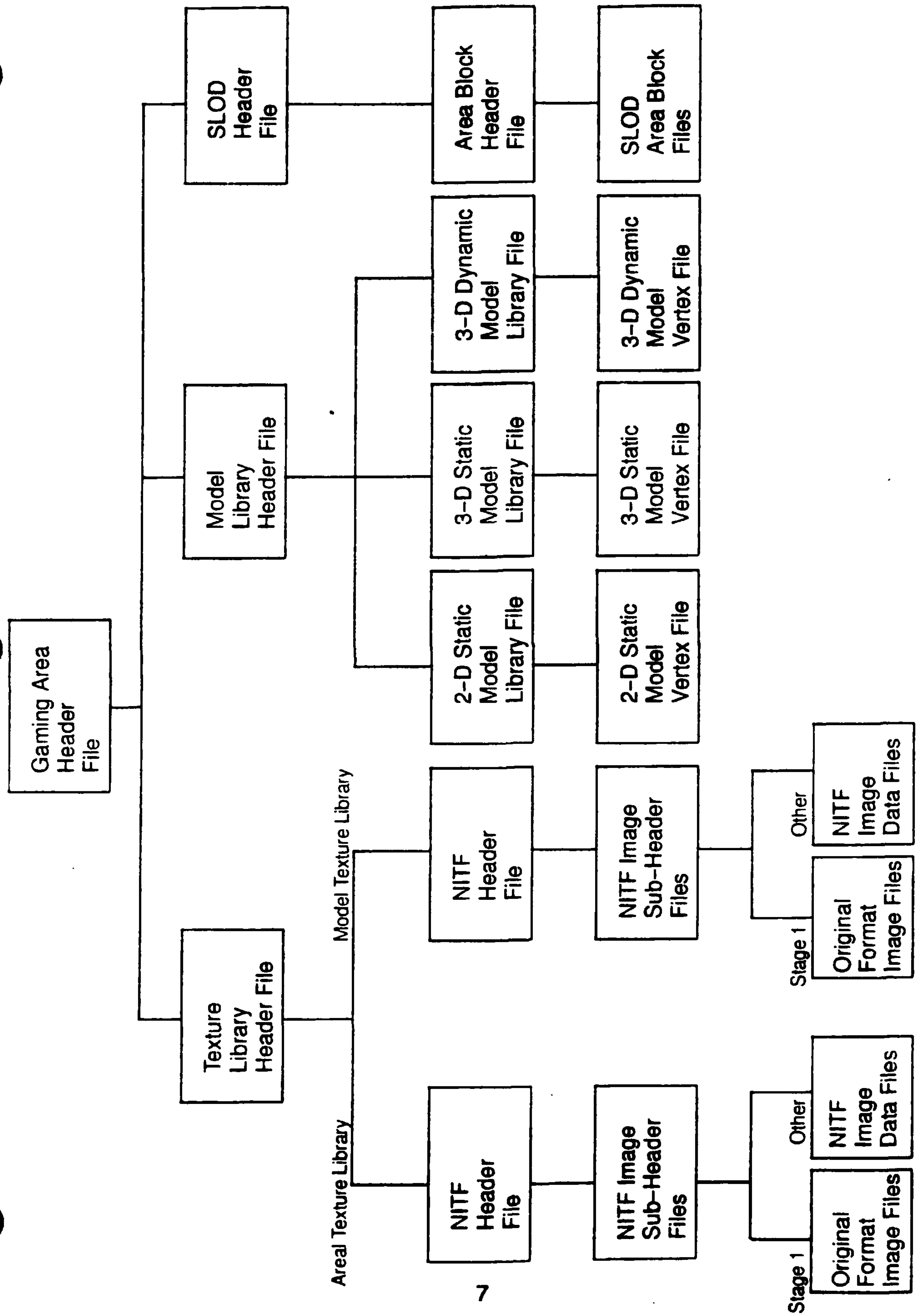


Figure 1. General Architecture of GTDB.

5.1.2 The following subparagraphs describe each of the GTDB files and the logical relationships among them. The file order on the GTDB tape shall be as follows:

```
Gaming Area Header (GAH) File
Model Library Header (MLH) File
SLOD Header (SLODH) File
Area Block Header (ABH) File
Texture Library Header (TLH) File
Two-Dimensional Static Model (2DSM) Library [optional]
Two-Dimensional Static Model Vertex (2DSMV) File [optional]
Three-Dimensional Static Model (3DSM) Library [optional]
Three-Dimensional Static Model Vertex (3DSMV) File [optional]
Three-Dimensional Dynamic Model (3DDM) Library [optional]
Three-Dimensional Dynamic Model Vertex (3DDMV) File [optional]

for each SLOD
   SLOD Area Blocks File

Areal Texture (AT) Library [optional]
   NITF Header File [optional]
   for each areal texture
      NITF Image Sub-Header File [optional]
      if texture is in Stage 1 then
         Original Format Image File(s) [optional]
      else
         NITF Image Data File [optional]
Model Texture (MT) Library [optional]
   NITF Header File [optional]
   for each model texture
      NITF Image Sub-Header File [optional]
      if texture is in Stage 1 then
         Original Format Image File(s) [optional]
      else
         NITF Image Data File [optional]
```

5.1.3 The record structure of each of these files shall be as described in the following sections.

5.2 **Gaming Area Header (GAH) File.** There shall be one Gaming Area Header File at the beginning of a GTDB.

5.2.1 **GAH Record Order.** The record order of the GAH file shall be as follows:

GAH Identifier Record
File Name Record
Gaming Area Header Record
GTDB Parameter Record
Boundary Point Records
Model List Record(s) [optional]
for each SLOD
   SLOD Parameter Record
      Keep-List Record(s) [optional]
      Delete-List Record(s) [optional]
      Level-List Record(s) [optional]
      for each area block
         Area Block Parameter Record
for each island
   Island Record
      Island LOD Record(s)
      Island Boundary Point Records
Option Record
Affected AB Count Record
for each Affected Area Block
   Affected AB ID Record
Checksum Record

5.2.2 **GAH Field Structure.** The field structure of each of these records shall be as described below.

5.2.2.1 **GAH Identifier Record.** This record shall consist of the ASCII string 'GAH'.

5.2.2.2 **GAH File Name Record.** This record shall consist of the ASCII string 'ssGnnnnnnGA.H', where "ss" is the security code and "nnnnnn" is the GTDB identifier.

5.2.2.3 **Gaming Area Header Record.** The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
GTDB Version Number Field
Last Update Date Field
Compilation Date Field
Security Level Field
Tape ID Field
Data Location Field
GTDB Directory Field

5.2.2.4 GTDB Parameters Record. The field structure of this record shall be as follows:

Coordinate System Parameters Subrecord
Boundary Point Count Field
Terrain Polygons Flag Field
Terrain Grid Flag Field
Terrain Grid Source Flag Field
Match Terrain At SLODs Flag Field
Vertex Normals Flag Field
Synthetic Culture Flag Field
Fragment Culture Flag Field
Decompose Culture Flag Field
Maximum Number of Edges Field
Use Models Flag Field
Decompose Models Flag Field
Maximum Number of Model Polygon Edges Field
Separation Planes Flag Field
Expand Lineals Flag Field
Fragment Point Light Strings Flag Field
Model List Count Field
SLOD Count Field (Always one or greater)
Island Count Field
Specific Areal Texture Parameters Subrecord
Generic Areal Texture Parameters Subrecord
Specific Model Texture Parameters Subrecord
Generic Model Texture Parameters Subrecord
User Option Field

5.2.2.4.1 Coordinate System Parameters Subrecord. The field structure of this record shall be as follows:

Coordinate System Field
Datum ID Field
Eccentricity Field
Semi-Major Axis Field
Datum Shift Field
Elevation Reference Field
Longitudinal Origin Field
Latitudinal Origin Field
Origin of Eastings Field
Origin of Northings Field
Scale Factor Field
First Standard Parallel Field
Second Standard Parallel Field
Tangency Point Height Field

5.2.2.4.2 **Specific Areal Texture Parameters Subrecord.** The field structure of this record shall be as follows:

Color Texture Existence Flag Field
Grayscale Texture Existence Flag Field
SMC/FDC Texture Existence Flag Field
Multispectral Texture Existence Flag Field
Processing Stage Field
Texture Format Field
Horizontal Block Size Field
Vertical Block Size Field
Number of Horizontal Blocks Field
Number of Vertical Blocks Field
Bits Per Texel Per Band Field
SMC/FDC Lookup-Table Existence Flag Field
Special Environmental Conditions Preference Field
Time Of Year Preference Subrecord
Image Capture Date Range Subrecord
Acceptable Percentage Of Cloud Cover Subrecord
Acceptable Percentage of Shadow Cover Subrecord

5.2.2.4.3 **Generic Areal Texture Parameters Subrecord.** The field structure of this record shall be as follows:

Color Texture Existence Flag Field
Grayscale Texture Existence Flag Field
Global-Based Mapping Flag Field
Face-Based Mapping Flag Field
Non-Mapped Flag Field
Texture Format Field
Horizontal Block Size Field
Vertical Block Size Field
Number of Horizontal Blocks Field
Number of Vertical Blocks Field
Bits Per Texel Per Band Field
Time Of Year Preference Subrecord

5.2.2.4.4 **Specific Model Texture Parameters Subrecord.** The field structure of this record shall be as follows:

Color Texture Existence Flag Field
Grayscale Texture Existence Flag Field
Multispectral Texture Existence Flag Field
Processing Stage Field
Model-Based Mapping Flag Field
Face-Based Mapping Flag Field
Vertex-to-Vertex Mapping Flag Field
Non-Mapped Flag Field
Texture Format Field
Horizontal Block Size Field
Vertical Block Size Field
Number of Horizontal Blocks Field
Number of Vertical Blocks Field
Bits Per Texel Per Band Field

5.2.2.4.5 **Generic Model Texture Parameters Subrecord.** The field structure of this record shall be as follows:

Color Texture Existence Flag Field
Grayscale Texture Existence Flag Field
Model-Based Mapping Flag Field
Face-Based Mapping Flag Field
Non-Mapped Flag Field
Texture Format Field
Horizontal Block Size Field
Vertical Block Size Field
Number of Horizontal Blocks Field
Number of Vertical Blocks Field
Bits Per Texel Per Band Field

5.2.2.4.6 **Time of Year Preference Subrecord.** The field structure of this record shall be as follows:

Spring Flag Field
Summer Flag Field
Autumn Flag Field
Winter Flag Field

5.2.2.4.7 **Image Capture Data Range Subrecord.** The field structure of this record shall be as follows:

Start Date Field
End Date Field

5.2.2.4.8 **Acceptable Percentage of Cloud Cover Subrecord.** The field structure of this record shall be as follows:

Low Value Field
High Value Field

5.2.2.4.9 **Acceptable Percentage of Shadow Cover Subrecord.** The field structure of this record shall be as follows:

Low Value Field
High Value Field

5.2.2.5 **Boundary Point Record.** The number of Boundary Point Records shall correspond to the value in the Boundary Point Count Field in the parent GTDB Parameters Record. The first and last boundary points shall be identical, and boundary points shall be sequenced in counterclockwise order as viewed from above. Edges defined by boundary points shall lie parallel to the axes of the coordinate plane. The field structure of this record shall be as follows:

Boundary Point Field

5.2.2.6 **Model List Record.** The number of these records shall correspond to the value in the Model List Count Field in the parent GTDB Parameters record. The field structure of each record shall be as follows:

Model Library Type Field
Model Number Field
Specific Model Texture Parameters Subrecord
Generic Model Texture Parameters Subrecord

5.2.2.6.1 **Specific Model Texture Parameters Subrecord.** The field structure of this record shall be as follows:

Color Texture Existence Flag Field
Grayscale Texture Existence Flag Field
Multispectral Texture Existence Flag Field
Processing Stage Field
Model-Based Mapping Flag Field
Face-Based Mapping Flag Field
Vertex-to-Vertex Mapping Flag Field
Non-Mapped Flag Field
Texture Format Field
Horizontal Block Size Field
Vertical Block Size Field
Horizontal Blocks Field
Vertical Blocks Field
Bits Per Texel Per Band Field

5.2.2.6.2 **Generic Model Texture Parameters Subrecord.** The field structure of this record shall be as follows:

Color Texture Existence Flag Field
Grayscale Texture Existence Flag Field
Model-Based Mapping Flag Field
Face-Based Mapping Flag Field
Non-Mapped Flag Field
Texture Format Field
Horizontal Block Size Field
Vertical Block Size Field
Number of Horizontal Blocks Field
Number of Vertical Blocks Field
Bits Per Texel Per Band Field

5.2.2.7 **SLOD Parameters Record.** The number of Simulator Level of Detail Parameters Records shall correspond to the value in the Simulator Level of detail Count Field in the parent GTDB Parameters Record. The field structure of this record shall be as follows:

Simulator Level of Detail ID Field
Number of Keep-List Entries Field
Number of Delete-List Entries Field
Number of Level-List Entries Field
Culture Resolution Field
Terrain Grid Source Simulator Level of Detail Field
Number of Area Blocks Field (Always one or greater)

5.2.2.8 **Keep-List Record.** The number of these records shall correspond to the value in the Number of Keep-List Entries field in the parent Simulator Level of Detail Parameters record. The Ending FDC Code Field shall always have a value equal to or greater than that of the starting FDC Code Field. The field structure of each record shall be as follows:

Starting FDC Code Field
Ending FDC Code Field

5.2.2.9 **Delete-List Record.** The number of these records shall correspond to the value in the Number of Delete-List Entries field in the parent Simulator Level of Detail Parameters record. The Ending FDC Code Field shall always have a value equal to or greater than that of the starting FDC Code Field. The field structure of each record shall be as follows:

Starting FDC Code Field
Ending FDC Code Field

5.2.2.10 **Level-List Record.** The number of these records shall correspond to the value in the Number of Level-List Entries field in the parent Simulator Level of Detail Parameters record. The Ending FDC Code Field shall always have a value equal to or greater than that of the starting FDC Code Field. The field structure of each record shall be as follows:

Starting FDC Code Field
Ending FDC Code Field

5.2.2.11 **Area Block Parameters Record.** The number of Area Block Parameters Records shall correspond to the value in the Number of Area Blocks Field in the parent Simulator Level of Detail Parameters Record. The field structure of this record shall be as follows:

Area Block Number Field
Area Block Boundary Field
Goodness-of-Fit Field
Minimum Number of Terrain Polygons Field
Maximum Number of Terrain Polygons Field
Maximum Number of Culture Polygons Field
Maximum Number of Model References Field
Terrain LOD Field
Color Texture Existence Flag Field
Grayscale Texture Existence Flag Field
SMC/FDC Texture Existence Flag Field
Color Texture Resolution Field
Grayscale Texture Resolution Field
SMC/FDC Texture Resolution Field

5.2.2.12 **Island Record.** The number of Island Records shall correspond to the value in the Island Count Field in the parent GTDB Parameters Record. The field structure of this record shall be as follows:

Island Number Field
Island Boundary Point Count Field

5.2.2.13 **Island LOD Record.** Following each Island Record, there shall be an Island LOD record for each simulator level of detail in the GTDB. The field structure of this record shall be as follows:

SLOD ID Field
SSDB Culture LOD Field
Color Texture Existence Flag Field
Grayscale Texture Existence Flag Field
Color Texture Resolution Field
Grayscale Texture Resolution Field

5.2.2.14 **Island Boundary Point Record.** The number of Island Boundary Point Records shall correspond to the value in the Island Boundary Point Count Field in the parent Island Record. The edge formed by successive Island Boundary Points shall not cross the boundary of any other island in the GTDB. The first and last boundary points of an island shall be identical, and boundary points shall be sequenced in counter-clockwise order as viewed from above. The field structure of this record shall be as follows:

Island Number Field
Boundary Point Field

5.2.2.15 **Option Record.** The field structure of the record shall be as follows:

Tape Option Field

5.2.2.16 **Affected AB Count Record.** Populated only if the Tape Option defined above indicates update only; otherwise, it shall contain zero. The field structure of the record shall be as follows:

Number of Area Blocks Field

5.2.2.17 **Affected AB ID Record.** The number of these records shall correspond to the Number of Area Blocks field in the Affected AB Count Record. The field structure of the record shall be as follows:

SLOD ID Field
Area Block Number Field

5.2.2.18 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.3 **Model Library Header (MLH) File.** There shall be one Model Library Header File following the TLH.

5.3.1 **MLH Record Order.** The record order of the MLH file shall be as follows:

MLH Identifier Record
File Name Record
for each model library
   Model Library Header Record
   Culture Color Table Record(s)
   Light Color Table Record(s)
   for each model
      Model LOD Complexity Table Record
      for each model LOD
         Model Complexity Statistics Table Subrecord
Checksum Record

5.3.2 **MLH Field Structure.** The field structure of each of these records shall be as described below.

5.3.2.1 **MLH Identifier Record.** This record shall consist of the ASCII string 'MLH'.

5.3.2.2 **File Name Record.** This record shall consist of the ASCII string 'ssGnnnnnnML.H', where "ss" is the security code and "nnnnnn" is the GTDB identifier.

5.3.2.3 **Model Library Header Record.** There shall be three Model Library Header records--one each for a 2-D Static Model Library, a 3-D Static Model Library, and a 3-D Dynamic Model Library. The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
Model Library Type Field
Last Update Date Field
Number of Culture Color Tables Field
Number of Light Color Tables Field
Number of Models Field

5.3.2.4 **Culture Color Table Record.** The total number of records shall correspond to the value in the "Number of Culture Color Tables" field in the parent Model Library Header record. The field structure of each record shall be as follows:

Color ID Field
Color Name Field
Color (Hue, Chroma, Value, Color Calibration Entry) Field
Number of Color References Field

5.3.2.5 **Light Color Table Record.** The total number of records shall correspond to the value in the "Number of Light Color Tables" field in the parent Model Library Header record. The field structure of each record shall be as follows:

Color ID Field
Color Name Field
Color (Hue, Chroma, Value, Color Calibration Entry) Field
Number of Color References Field

5.3.2.6 **Model LOD Complexity Table Record.** The total number of these records shall correspond to the value in the "Number of Models" field in the parent Model Library Header record. The field structure of each record shall be as follows:

Model Number Field
Number of LODs Field

5.3.2.7 **Model Complexity Statistics Table Record.** The total number of these records shall correspond to the value in the "Number of LODs" field in the parent Model LOD Complexity Table record. The field structure of each record shall be as follows:

Model Number Field
Model LOD Field
Number of Polygons Field
Number of Separation Planes Field
Number of Texture References Field
Number of Collision Test Points Field
Correlation Priority Field

5.3.2.8 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.4 **Simulator Level of Detail Header (SLODH) File.** There shall be one Simulator Level of Detail Header File following the MLH.

5.4.1 **SLODH Record Order.** The record order of the SLODH file shall be as follows:

SLODH Identifier Record
File Name Record
SLODH File Header Record
for each SLOD
  SLOD Header Record
  Feature Distribution Table Record
  Z-Density Distribution Table Record
  SMC Distribution Table Record
  Culture Color Table Record
  Light Color Table Record
Checksum Record

5.4.2 **SLODH Field Structure.** The field str''cture of each of these records shall be as described below.

5.4.2.1 **SLODH Identifier Record.** This record shall consist of the ASCII string 'SLODH'.

5.4.2.2 **File Name Record.** This record shall consist of the ASCII string 'ssGnnnnnnSLOD.H', where "ss" is the security code and "nnnnnn" is the GTDB identifier.

5.4.2.3 **SLODH File Header Record.** This record shall consist of a single field indicating the number of SLODs defined within the GTDB, as follows:

Number of SLODs Field

5.4.2.4 **SLOD Header Record.** The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
SLOD ID Field
Area Block Size Field
Last Update Date Field
Security Level Field
SLOD Polygon Density Statistics Table Subrecord
Number of Feature Distribution Tables Field
Number of Z-Density Distribution Tables Field
Number of SMC Distribution Tables Field
Number of Culture Color Tables Field
Number of Light Color Tables Field

5.4.2.4.1 **SLOD Polygon Density Statistics Table Subrecord.** The field structure of this record shall be as follows:

Maximum Terrain Polygons Field
Minimum Terrain Polygons Field
Maximum Areal Feature Polygons Field
Minimum Areal Feature Polygons Field
Maximum Linear Feature Segments Field
Minimum Linear Feature Segments Field
Maximum Point Features Field
Minimum Point Features Field
Maximum Point Light Features Field
Minimum Point Light Features Field
Maximum Point Light Strings Field
Minimum Point Light Strings Field
Maximum Model References Field
Minimum Model References Field
Maximum Texture References Field
Minimum Texture References Field
Maximum Model Polygons Field
Minimum Model Polygons Field
Maximum Total Elements Field
Minimum Total Elements Field

5.4.2.5 **Feature Distribution Table Record.** The total number of records shall correspond to the value in the "Number of Feature Distribution Tables" field in the parent Simulator Level of Detail Header record. The field structure of each record shall be as follows:

Feature Descriptor Code Field
Number of Feature Occurrences Field
Number of Fragments Field
Correlation Priority Field

5.4.2.6 **Z-Density Distribution Table Record.** The total number of records shall correspond to the value in the "Number of Z-Density Distribution Tables" field in the parent Simulator Level of Detail Header record. The field structure of each record shall be as follows:

Number of Layers Above Terrain Polygon Field
Number of Occurrences Field

5.4.2.7 **SMC Distribution Record.** The total number of records shall correspond to the value in the "Number of SMC Distribution Tables" field in the parent Simulator Level of Detail Header record. The field structure of each record shall be as follows:

Surface Material Category Field
Surface Material Subtype Field
Number of Occurrences Field

5.4.2.8 **Culture Color Table Record.** The total number of records shall correspond to the value in the "Number of Culture Color Tables" field in the parent Simulator Level of Detail Header record. The field structure of each record shall be as follows:

Color ID Field
Color Name Field
Color (Hue, Chroma, Value, Color Calibration Entry) Field
Number of Color References Field

5.4.2.9 **Light Color Table Record.** The total number of records shall correspond to the value in the "Number of Light Color Tables" field in the parent Simulator Level of Detail Header record. The field structure of each record shall be as follows:

Color ID Field
Color Name Field
Color (Hue, Chroma, Value, Color Calibration Entry) Field
Number of Color References Field

5.4.2.10 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.5 **Area Block Header (ABH) File.** There shall be one Area Block Header File following the SLODH.

5.5.1 **ABH Record Order.** The record order of the ABH file shall be as follows:

ABH Identifier Record
File Name Record
for each SLOD
  ABH File Header Record
  for each area block within SLOD[4]
    Area Block Header Record
    Feature Distribution Table Record
    SMC Distribution Table Record
    Culture Color Table Record
    Light Color Table Record
    Areal Texture Table Record
Checksum Record

5.5.2 **ABH Field Structure.** The field structure of each of these records shall be as described below.

5.5.2.1 **ABH Identifier Record.** This record shall consist of the ASCII string 'ABH'.

5.5.2.2 **File Name Record.** This record shall consist of the ASCII string 'ssGnnnnnnAB.H', where "ss" is the security code and "nnnnnn" is the GTDB identifier.

5.5.2.3 **ABH File Header Record.** This record shall consist of a single field indicating the number of area blocks defined within the SLOD, as follows:

Number of Area Blocks Field

5.5.2.4 **Area Block Header Record.** The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
SLOD ID Field
Area Block Number Field
Lat/Long SW Corner Field
Lat/Long NE Corner Field
Last Update Date Field
Security Level Field
Polygon Density Statistics Table Subrecord
Terrain Roughness Statistics Table Subrecord
Area Block Existence Flags Subrecord
Number of Feature Distribution Tables Field
Number of SMC Distribution Tables Field
Number of Culture Color Tables Field
Number of Light Color Tables Field

5.5.2.4.1 **Polygon Density Statistics Table Subrecord.** The field structure of this subrecord is as follows:

Number of Vertices Field
Number of Terrain Polygons Field
Number of Areal Feature Polygons Field
Number of Linear Feature Segments Field
Number of Point Features Field
Number of Point Light Features Field
Number of Point Light Strings Field
Number of Model References Field
Number of Texture References Field

5.5.2.4.2 **Terrain Roughness Statistics Subrecord.** These fields shall be populated when gridded terrain has been requested, and zero otherwise. The field structure of this record shall be as follows:

Maximum Elevation Field
Minimum Elevation Field
Terrain Roughness Index Field

5.5.2.4.3 **Area Block Existence Flags Subrecord.** The field structure of this record shall be as follows:

Areal Feature Area Block Flag (Always True)
Linear Feature Area Block Flag
Point Feature Area Block Flag
Point Light Feature Area Block Flag
Point Light String Feature Area Block Flag
Terrain Polygon Area Block Flag
Terrain Grid Area Block Flag
Model Reference Area Block Flag

5.5.2.5 **Feature Distribution Table Record.** The total number of records shall correspond to the value in the "Number of Feature Distribution Tables" field in the parent Area Block Header record. The field structure of each record shall be as follows:

Feature Descriptor Code Field
Number of Feature Occurrences Field
Number of Fragments Field
Correlation Priority Field

5.5.2.6 **SMC Distribution Record.** The total number of records shall correspond to the value in the "Number of SMC Distribution Tables" field in the parent Area Block Header record. The field structure of each record shall be as follows:

Surface Material Category Field
Surface Material Subtype Field
Number of Occurrences Field

5.5.2.7 **Culture Color Table Record.** The total number of records shall correspond to the value in the "Number of Culture Color Tables" field in the parent Area Block Header record. The field structure of each record shall be as follows:

Color ID Field
Color Name Field
Color (Hue, Chroma, Value, Color Calibration Entry) Field
Number of Color References Field

5.5.2.8 **Light Color Table Record.** The total number of records shall correspond to the value in the "Number of Light Color Tables" field in the parent Area Block Header record. The field structure of each record shall be as follows:

Color ID Field
Color Name Field
Color (Hue, Chroma, Value, Color Calibration Entry) Field
Number of Color References Field

5.5.2.9 **Areal Texture Table Record.** The total number of records shall correspond to the value in the "Number of Texture References" field in the Polygon Density Statistics Table Subrecord within the parent Area Block Header record. The field structure of each record shall be as follows:

GTDB Texture Library Type Field
Texture ID Field

5.5.2.10 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.6 **Texture Library Header (TLH) File.** There shall be one Texture Library Header File following the GAH.

5.6.1 **TLH Record Order.** The record order of the TLH file shall be as follows:

TLH Identifier Record
File Name Record
for each texture library (2)
   Texture Library Complexity Statistics Record
   Texture Library Header Record
   Texture Distribution Table Record(s)
   Stage 1 Texture File Association Record(s)
Checksum Record

5.6.2 **TLH Field Structure.** The field structure of each of these records shall be as described below.

5.6.2.1 **TLH Identifier Record.** This record shall consist of the ASCII string 'TLH'.

5.6.2.2 **File Name Record.** This record shall consist of the ASCII string 'ssGnnnnnnTL.H', where "ss" is the security code and "nnnnnn" is the GTDB identifier.

5.6.2.3 **Texture Library Complexity Statistics Record.** There shall be two Texture Library Complexity Statistics records, one for an Areal Texture Library and one for a Model Texture Library, whether or not those optional libraries actually exist within the GTDB. The "GTDB Texture Library Type" field shall indicate which of the two libraries is being described. A value of zero in the "Number of Texture Images in Library" field shall indicate that there is no actual texture library of the given type within the GTDB. The field structure shall be as follows:

P2851 GTDB Catalog ID Field
GTDB Texture Library Type Field
Number of Texture Images in Library Field
Number of Stage 1 Texture File Associations Field
Number of Stage 1 Specific Textures Field
Stage 1 Specific Textures Storage Size Field
Number of Stage 2 Specific Textures Field
Stage 2 Specific Textures Storage Size Field
Number of Stage 3 Specific Textures Field
Stage 3 Specific Textures Storage Size Field
Number of Stage 4 Specific Textures Field
Stage 4 Specific Textures Storage Size Field
Number of Stage 5 Specific Textures Field
Stage 5 Specific Textures Storage Size Field
Number of Stage 3 Generic Textures Field
Stage 3 Generic Textures Storage Size Field
Number of Stage 4 Generic Textures Field
Stage 4 Generic Textures Storage Size Field
Number of Stage 5 Generic Textures Field
Stage 5 Generic Textures Storage Size Field
Number of Stage 3 SMC/FDC Textures Field
Stage 3 SMC/FDC Textures Storage Size Field
Number of Stage 4 SMC/FDC Textures Field
Stage 4 SMC/FDC Textures Storage Size Field
Number of Stage 5 SMC/FDC Textures Field
Stage 5 SMC/FDC Textures Storage Size Field

5.6.2.4 **Texture Library Header Record.** There shall be two Texture Library Header records, one for an Areal Photo Texture Library and one for a Model Photo Texture Library. The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
GTDB Texture Library Type Field
Security Level Field
Last Update Date Field

5.6.2.5 **Texture Distribution Table Record.** The number of these records shall correspond to the value in the "Number of Texture Images in Library" field in the parent Texture Library Header record. The field structure of each record shall be as follows:

Texture ID Field
Processing Stage Field
Specific or Generic Texture Flag Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Storage Size Field
Texture Data Format Field
Number of Data Files Field

5.6.2.6 **Stage 1 Texture Field Association Record.** The number of these records shall be zero for all non-Stage 1 textures and (Number of Data Files - 1) for all Stage 1 textures. The field structure of each record shall be as follows:

Texture ID Field
File Name Field
Original File Name Field

5.6.2.7 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.7 **2-D Static Model (2DSM) Library File.** There may be one 2-D Static Model Library File following the MPT. The 2DSM is optional in a GTDB.

5.7.1 **2DSM Record Order.** When included, the record order of the 2DSM file shall be as follows:

```
2DSM Identifier Record
File Name Record
2DSM (Model Library) Header Record
for each model
   Model Header Record
   for each model LOD
      LOD Header Record
      LOD Texture Reference Pointer Record(s) [optional]
      for each component
         Component Header Record
         Component Texture Reference Pointer Record(s) [optional]
         for each model polygon
            Model Polygon Record
            Microdescriptor Record(s) [optional]
            Vertex Pointer Records
            Polygon FACS Record(s) [optional]
            Polygon Texture Reference Pointer Record(s) [optional]
      Subsidiary Model References Record(s) [optional]
      for each point light string
         Point Light String Record(s) [optional]
         Point Light String FACS Record(s) [optional]
      Model FACS Record(s) [optional]
      Face-Based Texture Reference Record(s) [optional]
      Vertex-to-Vertex Texture Reference Record(s) [optional]
      Model-Based Texture Reference Record(s) [optional]
      Non-Mapped Texture Reference Record(s) [optional]
Checksum Record
```

5.7.2 **2DSM Field Structure.** The field structure of each of these records shall be as described below.

5.7.2.1 **2DSM Identifier Record.** This record shall consist of the ASCII string '2DSM'.

5.7.2.2 **File Name Record.** This record shall consist of the ASCII string 'ssGnnnnnn2DS.LIB', where "ss" is the security code, and "nnnnnn" is the GTDB identifier.

5.7.2.3 **2DSM Header Record.** The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
Model Library Type Field
Security Level Field
Last Update Date Field
Number of Models Field

5.7.2.4 **Model Header Record.** The number of these records shall correspond to the value contained in the Number of Models field in the parent 2DSM Header record. The field structure of this record shall be as follows:

Model Number Field
Model Name Field
Model Description Field
Generic Model Flag Field
Feature Descriptor Code Field
Number of Model LODs Field

5.7.2.5 **LOD Header Record.** The number of these records for a given model group shall correspond to the value contained in the Number of Model LODs field in the parent Model Header record. The field structure of this record shall be as follows:

Model Number Field
Model LOD Field
LOD Resolution Description Field
Sensor Types Supported Field
Source Simulator Field
Directivity Field
Radius Field
Predominant Height Field
Centroid Field
Base Polygon ID Field
Percentage of Texture Coverage Field
Number of Polygons Field
Number of LOD Texture Reference Pointers Field
Number of Components Field
Number of Subsidiary Model References Field
Number of Point Light Strings Field
Number of Model FACS Field
Number of Face-Based Texture References Field
Number of Vertex-to-Vertex Texture References Field
Number of Model-Based Texture References Field
Number of Non-Mapped Texture References Field
Number of Separation Planes Field [Always Zero]
Number of Collision Test Points Field [Always Zero]

5.7.2.6 **LOD Texture Reference Pointer Record.** The number of these records for a given model LOD shall correspond to the value contained in the Number of LOD Texture Reference Pointers Field in the parent LOD Header record. The field structure of this record shall be as follows:

Texture Mapping Type Field
Texture Reference e Table Index Field
Texture Mapping Set ID Field

5.7.2.7 **Component Header Record.** The number of these records for a given model LOD shall correspond to the value contained in the Number of Components field in the parent LOD Header record. The field structure of this record shall be as follows:

Component ID Field
Number of Component Texture Reference Pointers Field
Number of Polygons Field

5.7.2.8 **Component Texture Reference Pointer Record.** The number of these records for a given component shall correspond to the value contained in the Number of Component Texture Reference Pointers field in the parent Component Header record. The field structure of this record shall be as follows:

Texture Mapping Type Field
Texture Reference Table Index Field
Texture Mapping Set ID Field

5.7.2.9 **Model Polygon Record**. The number of these records for a given component shall correspond to the value contained in the Number of Polygons field in the parent Component Header Record. The total number of these records for each model LOD shall correspond to the Number of Polygons field in the LOD Header Record. Each model polygon shall belong to exactly one component. The field structure of this record shall be as follows:

Polygon ID Field
Cluster ID Field
Component ID Field
Surface Material Category Field
Surface Material Subtype Field
Reflectance Field
Light Type Field
Specular Field
Polygon Non-Shadow Field
Polygon Normal Field
Transmissivity Field
Polygon Long Dimension Field
Polygon Short Dimension Field
Centroid Field
Diffuse Reflectance Field
Feature Onset Field
Layer Number (Radar) Field
Color Characteristics Field
Shading Type Field
Translucency Field
Polygon Non-Occulting Field
Cycle Rate Off Field
Cycle Rate On Field
Directionality Field
Light Horizontal Width Field
Light Horizontal Center Field
Light Horizontal Fall Field
Light Intensity Field
Light Vertical Width Field
Light Vertical Center Field
Light Vertical Fall Field
Polygon Illumination Type Field
Polygon Landing Light Illumination Field
Absorptivity Field
Emissivity Field
Exitance Field
Self-Emitter Field
Layer Number (Visual) Field
Layer Number (Infrared) Field
Number of Microdescriptors Field
Number of Vertices Field (Always three or greater)
Number of Polygon Texture Reference Pointers Field
Number of Polygon FACS Field

5.7.2.10 **Model Microdescriptor Record**. The number of these records shall correspond to the value in the Number of Microdescriptors field in the parent Model Polygon record. The field structure of each record shall be as follows:

Microdescriptor Type Field
Microdescriptor Value Field

5.7.2.11 **Vertex Pointer Record.** The number of records shall correspond to the Number of Vertices field within the parent Model Polygon record. Polygons shall be closed implicitly, i.e., the first vectex shall not be repeated as the last. The field structure of each record shall be as follows:

Vertex List Position Field
Normal List Position Field [Always Zero]
Correlation Priority Field

5.7.2.12 **Polygon FACS Record.** The number of these records shall correspond to the value in the Number of Polygon FACS field in the parent Model Polygon record. The field structure of each record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Length of Attribute Field
Attribute Value Field

5.7.2.13 **Polygon Texture Reference Pointer Record.** The number of these records shall correspond to the value of the Number of Polygon Texture Reference Pointer Field within the parent Model Polygon record. The field structure of each record shall be as follows:

Texture Mapping Type Field
Texture Reference Table Index Field
Texture Mapping Set ID Field

5.7.2.14 **Subsidiary Model Reference Record.** The number of these records for a given model shall correspond to the value contained in the Number of Subsidiary Model References field in the parent Model record. The field structure of this record shall be as follows:

Referenced Model Library Type Field
Referenced Model Number Field
Referenced Model LOD Field
Translation Field
Scale Factor Field
Rotation Angles Field
Articulated Part Flag Field

5.7.2.15 **Point Light String Record.** The number of Point Light String records shall correspond to the value in the Number of Point Light Strings field within the LOD Header record. The field structure of this record shall be as follows:

Length Field
Orientation Field
Shape Code Field
Width Field
Directionality Field
Light Type Field
Predominant Height Field
Surface Material Category Field
Color Characteristics Field
Layer Number Field
Absorptivity Field
Centroid Field
Cycle Rate Off Time Field
Cycle Rate On Time Field
Diffuse Reflectance Field
Directivity (Infrared) Field
Directivity (Radar) Field
Directivity Field
Emissivity Field
Exitance Field
Feature Onset Field
Internal Material Category Field
Internal Material Volume Field
Layer Number (Infrared) Field
Light Horizontal Center Field
Light Horizontal Fall Field
Light Horizontal Width Field
Light Intensity Field
Light Vertical Center Field
Light Vertical Fall Field
Light Vertical Width Field
Long Lineal Field
Low Level Effects Field
Object Volume Field
Radius Field
Reflectance Field
Self-Emitter Field
Surface Material Subtype Field
Texture Map Reflectance Field
Transmissivity Field
Visible Range Field
Number of FACS Table Entries Field
Number of Lights Field
for each light in the string
   Position Field

5.7.2.16 **Point Light String FACS Record.** The number of these records shall correspond to the value in the Number of FACS Table Entries field in the parent Point Light String Record. The field structure of each record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Length of Attribute Field
Attribute Value Field

5.7.2.17 **Model FACS Record.** The number of these records shall correspond to the value in the Number of Model FACS field in the parent Model record. The field structure of each record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Length of Attribute Field
Attribute Value Field

5.7.2.18 **Face-Based Texture Reference Record.** The number of these records shall correspond to the value of the Number of Face-Based Texture References field in the parent LOD Header Record. The field structure of this record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific Or Generic Texture Flag Field
Texture Origin Field
Boundary ID Field
Mirror Field
Wrap Field
Wrap Type Field
Texture Scale Field
Polygon Alignment Vector Field
Rotation About Texture Origin Field
Polygon Reference Point Field
Layer Number Field

5.7.2.19 **Vertex-to-Vertex Texture Reference Record.** The number of these records shall correspond to the value of the Number of Vertex-to-Vertex Texture References field in the parent LOD Header Record. The field structure of this record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific Or Generic Texture Flag Field
Layer Number Field
Number of Texture Pattern Coordinates Field
for each texture pattern vertex
    Texture Pattern Coordinates Field

5.7.2.20 **Model-Based Texture Reference Record.** The number of these records shall correspond to the value of the Number of Model-Based Texture References field in the parent LOD Header Record. The field structure of this record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific Or Generic Texture Flag Field
Texture Origin Field
Boundary ID Field
Mirror Field
Wrap Field
Wrap Type Field
Texture Scale Field
Orientation Vectors Field
Model Reference Point Field
Layer Number Field

5.7.2.21 **Non-Mapped Texture Reference Record.** The number of these records shall correspond to the value contained in the Number of Non-Mapped Texture References field in the parent Model Header record. The field structure of this record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific Or Generic Texture Flag Field

5.7.2.22 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.8 **2-D Static Model Vertex (2DSMV) File.** Following the 2DSM file, there shall be one 2-D Static Model Vertex File. The 2DSMV shall be required if there is a 2DSM, but shall be omitted otherwise.

5.8.1 **2DSMV File Structure.** There shall be one pseudo-file for each model, containing the vertices used to define all the LODs of that model. These pseudo-files shall physically occur on the tape in the same sequence as their corresponding model definitions occur within the model library. Each pseudo-file shall be terminated by a special record indicating a pseudo-EOF (end of file). The pseudo-file structure within this file shall be as follows:

for each model in 2DSM
    2DSMV Pseudo-File

5.8.2 **2DSMV Record Structure.** The record structure of each 2DSMV pseudo-file shall be as defined in the following subsections.

5.8.2.1 **2-D Static Model Vertex Pseudo-Files.** There shall be one pseudo-file for each model, containing the vertices used to define all the LODs of that model.

5.8.2.1.1 **2DSMV Pseudo-File Record Order.** The record order of each 2DSMV pseudo-file shall be as follows:

2DSMV Identifier Record
File Name Record
for each model vertex in 2DSM
  Vertex Record
Pseudo-EOF Record
Checksum Record

5.8.2.1.2 **2DSMV Pseudo-File Field Structure.** The field structure of each of these records shall be as described below.

5.8.2.1.2.1 **2DSMV Identifier Record.** This record shall consist of the ASCII string '2DSMV'.

5.8.2.1.2.2 **File Name Record.** This record shall consist of the ASCII string 'ssGnnnnnn2DS.VTX', where "ss" is the security code, and "nnnnnn" is the GTDB identifier.

5.8.2.1.2.3 **Vertex Record.** The field structure of this record shall be as follows:

Coordinate Field

5.8.2.1.2.4 **Pseudo-EOF Record.** This record shall consist of the ASCII string 'EOF MDL2DSnnnnn.VTX', where 'nnnnn' is the number of the model described by this pseudo-file.

5.8.2.1.2.5 **Checksum Record.** The field structure of this record is as follows:

Checksum Field

5.9 **3-D Static Model (3DSM) Library File.** There may be one 3-D Static Model Library File following the 2DSMV. The 3-D Static Model Library File is optional in a GTDB.

5.9.1 **3DSM Record Order.** When included, the record order of the 3DSM file shall be as follows:

```
3DSM Identifier Record
File Name Record
3DSM (Model Library) Header Record
for each model
   Model Header Record
   for each model LOD
      LOD Header Record
      LOD Texture Reference Pointer Record(s) [optional]
      for each component
         Component Header Record
         Component Texture Reference Pointer Record(s) [optional]
         for each model polygon
            Model Polygon Record
            Microdescriptor Record(s) [optional]
            Vertex Pointer Records
            Polygon FACS Records [optional]
            Polygon Texture Reference Pointer Record(s) [optional]
      Subsidiary Model References Record(s) [optional]
      for each point light string
         Point Light String Record(s) [optional]
         Point Light String FACS Record(s) [optional]
      Model FACS Record(s) [optional]
      Face-Based Texture Reference Record(s) [optional]
      Vertex-to-Vertex Texture Reference Record(s) [optional]
      Model-Based Texture Reference Record(s) [optional]
      Non-Mapped Texture Reference(s) [optional]
      Separation Plane Record(s) [optional]
Checksum Record
```

5.9.2 **3DSM Field Structure.** The field structure of each of these records shall be as described below.

5.9.2.1 **3DSM Identifier Record.** This record shall consist of the ASCII string '3DSM'.

5.9.2.2 **3DSM File Name Record.** This record shall consist of the ASCII string 'ssGnnnnnn3DS.LIB', where "ss" is the security code, and "nnnnnn" is the GTDB identifier.

5.9.2.3 **3DSM Header Record.** The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
Model Library Type Field
Security Level Field
Last Update Date Field
Number of Models Field

5.9.2.4 Model Header Record. The number of these records shall correspond to the value contained in the Number of Models field in the parent 3DSM Header record. The field structure of this record shall be as follows:

Model Number Field
Model Name Field
Model Description Field
Generic Model Flag Field
Feature Descriptor Code Field
Number of Model LODs Field

5.9.2.5 LOD Header Record. The number of these records for a given model group shall correspond to the value contained in the Number of Model LODs field in the parent Model Header record. The field structure of this record shall be as follows:

Model Number Field
Model LOD Field
LOD Resolution Description Field
Sensor Types Supported Field
Source Simulator Field
Directivity Field
Radius Field
Predominant Height Field
Centroid Field
Base Polygon ID Field
Percentage of Texture Coverage Field
Number of Polygons Field
Number of LOD Texture Reference Pointers Field
Number of Components Field
Number of Subsidiary Model References Field
Number of Point Light Strings Field
Number of Model FACS Field
Number of Face-Based Texture References Field
Number of Vertex-to-Vertex Texture References Field
Number of Model-Based Texture References Field
Number of Non-Mapped Texture References Field
Number of Separation Planes Field
Number of Collision Test Points Field [Always Zero]

5.9.2.6 LOD Texture Reference Pointer Record. The number of these records for a given model LOD shall correspond to the value contained in the Number of LOD Texture Reference Pointers field in the parent LOD Header record. The field structure of this record shall be as follows:

Texture Mapping Type Field
Texture Reference Table Index Field
Texture Mapping Set ID Field

5.9.2.7 **Component Header Record.** The number of these records for a given model LOD shall correspond to the value contained in the Number of Components field in the parent LOD Header record. The field structure of this record shall be as follows:

Component ID Field
Number of Component Texture Reference Pointers Field
Number of Polygons Field

5.9.2.8 **Component Texture Reference Pointer Record.** The number of these records for a given model LOD shall correspond to the value contained in the Number of Component Texture Reference Pointers field in the parent Component Header record. The field structure of this record shall be as follows:

Texture Mapping Type Field
Texture Reference Table Index Field
Texture Mapping Set ID Field

5.9.2.9 **Model Polygon Record.** The number of these records for a given model shall correspond to the value contained in the Number of Polygons field in the parent Component Header record. The field structure of this record shall be as follows:

Polygon ID Field
Cluster ID Field
Component ID Field
Surface Material Category Field
Surface Material Subtype Field
Reflectance Field
Light Type Field
Specular Field
Polygon Non-Shadow Field
Polygon Normal Field
Transmissivity Field
Polygon Long Dimension Field
Polygon Short Dimension Field
Centroid Field
Diffuse Reflectance Field
Feature Onset Field
Layer Number (Radar) Field
Color Characteristics Field
Shading Type Field
Translucency Field
Polygon Non-Occulting Field
Cycle Rate On Field
Cycle Rate Off Field
Cycle Rate Field
Directionality Field
Light Horizontal Width Field
Light Horizontal Center Field
Light Horizontal Fall Field
Light Intensity Field
Light Vertical Width Field
Light Vertical Center Field
Light Vertical Fall Field
Polygon Illumination Type Field
Polygon Landing Light Illumination Field
Absorptivity Field
Emissivity Field
Exitance Field
Self-Emitter Field
Layer Number (Visual) Field
Layer Number (Infrared) Field
Number of Microdescriptors Field
Number of Vertices Field (Always three or greater)
Number of Polygon Texture Reference Pointers Field
Number of Polygon FACS Field

5.9.2.10 **Model Microdescriptor Record.** The number of these records shall correspond to the value in the Number of Microdescriptors field in the parent Model Polygon record. The field structure of each record shall be as follows:

Microdescriptor Type Field
Microdescriptor Value Field

5.9.2.11 **Vertex Pointer Record.** The number of records shall correspond to the Number of Vertices field within the parent Model Polygon record. Polygons shall be closed implicitly, i.e., the first vertex shall not be repeated as the last. The Normal List Position Field is zero when vertex normals have not been requested. The field structure of each record shall be as follows:

Vertex List Position Field
Normal List Position Field
Correlation Priority Field

5.9.2.12 **Polygon FACS Record.** The number of these records shall correspond to the value in the Number of Polygon FACS field in the parent Model Polygon record. The field structure of each record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Length of Attribute Field
Attribute Value Field

5.9.2.13 **Polygon Texture Reference Pointer Record.** The number of these records for a given model polygon shall correspond to the value of the Number of Polygon Texture Reference Pointer field within the parent Model Polygon record. The field structure of each record shall be as follows:

Texture Mapping Type Field
Texture Reference Table Index Field
Texture Mapping Set ID Field

5.9.2.14 **Subsidiary Model Reference Record.** The number of these records for a given model shall correspond to the value contained in the Number of Subsidiary Model References field in the parent Model record. The field structure of this record shall be as follows:

Referenced Model Library Type Field
Referenced Model Number Field
Referenced Model LOD Field
Translation Field
Scale Factor Field
Rotation Angles Field
Articulated Part Flag Field

5.9.2.15 **Point Light String Record.** The number of Point Light String records shall correspond to the value in the Number of Point Light Strings field within the LOD Header record. The field structure of this record shall be as follows:

Length Field
Orientation Field
Shape Code Field
Width Field
Directionality Field
Light Type Field
Predominant Height Field
Surface Material Category Field
Color Characteristics Field
Layer Number Field
Absorptivity Field
Centroid Field
Cycle Rate Off Time Field
Cycle Rate On Time Field
Diffuse Reflectance Field
Directivity (Infrared) Field
Directivity (Radar) Field
Directivity Field
Emissivity Field
Exitance Field
Feature Onset Field
Internal Material Category Field
Internal Material Volume Field
Layer Number (Infrared) Field
Light Horizontal Center Field
Light Horizontal Fall Field
Light Horizontal Width Field
Light Intensity Field
Light Vertical Center Field
Light Vertical Fall Field
Light Vertical Width Field
Long Lineal Field
Low Level Effects Field
Object Volume Field
Radius Field
Reflectance Field
Self-Emitter Field
Surface Material Subtype Field
Texture Map Reflectance Field
Transmissivity Field
Visible Range Field
Number of FACS Table Entries Field
Number of Lights Field
for each light in the string
  Position Field

5.9.2.16 **Point Light String FACS Record.** The number of these records shall correspond to the value in the Number of FACS Table Entries field in the parent Point Light String Record. The field structure of each record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Length of Attribute Field
Attribute Value Field

5.9.2.17 **Model FACS Record.** The number of these records shall correspond to the value in the Number of Model FACS field in the parent Model record. The field structure of each record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Length of Attribute Field
Attribute Value Field

5.9.2.18 **Face-Based Texture Reference Record.** The number of these records shall correspond to the value of the Number of Face-Based Texture References field in the parent LOD Header Record. The field structure of this record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific Or Generic Texture Flag Field
Texture Origin Field
Boundary ID Field
Mirror Field
Wrap Field
Wrap Type Field
Texture Scale Field
Polygon Alignment Vector Field
Rotation About Texture Origin Field
Polygon Reference Point Field
Layer Number Field

5.9.2.19 **Vertex-to-Vertex Texture Reference Record.** The number of these records shall correspond to the value of the Number of Vertex-to-Vertex Texture References field in the parent LOD Header Record. The field structure of this record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific Or Generic Texture Flag Field
Layer Number Field
Number of Texture Pattern Coordinates Field
for each texture pattern vertex
    Texture Pattern Coordinates Field

5.9.2.20 **Model-Based Texture Reference Record.** The number of these records shall correspond to the value of the Number of Model-Based Texture References field in the parent LOD Header Record. The field structure of this record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific Or Generic Texture Flag Field
Texture Origin Field
Boundary ID Field
Mirror Field
Wrap Field
Wrap Type Field
Texture Scale Field
Orientation Vectors Field
Model Reference Point Field
Layer Number Field

5.9.2.21 **Non-Mapped Texture Reference Record.** The number of these records shall correspond to the value contained in the Number of Non-Mapped Texture References field in the parent LOD Header record. The field structure of this record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific Or Generic Texture Flag Field

5.9.2.22 **Separation Plane Record.** The number of separation plane records for a given model shall correspond to the value contained in the Number of Separation Planes field in the parent LOD Header record. The separation plane records shall be stored in the order in which they were defined for the model. The field structure of this record shall be as follows:

Polygon ID Field
Separation Plane Number Field
Separation Plane Coefficients Field

5.9.2.23 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.10 **3-D Static Model Vertex (3DSMV) File.** Following the 3DSM file, there shall be one 3-D Static Model Vertex File. The 3DSMV shall be required if there is a 3DSM, but shall be omitted otherwise.

5.10.1 **3DSMV File Structure.** There shall be one pseudo-file for each model, containing the vertices used to define all the LODs of that model. Each pseudo-file shall be terminated by a special record indicating a pseudo-EOF (end of file). The pseudo-file structure within this file is as follows:

for each model in the 3DSM
   3DSMV Pseudo-File

5.10.2 **3DSMV Record Structure.** The record structure of each 3DSMV pseudo-file shall be as defined in the following subsections.

5.10.2.1 **3DSMV Pseudo-Files.** There shall be one pseudo-file for each model, containing the vertices used to define all the LODs of that model. The pseudo-files shall physically occur on tape in the same sequence as their corresponding model definitions occur within the model library.

5.10.2.1.1 **3DSMV Pseudo-File Record Order.** The record order of each 3DSMV pseudo-file shall be as follows:

    3DSMV Identifier Record
    File Name Record
    for each model vertex and vertex normal in 3DSM
       Vertex Record
    Pseudo-EOF Record
    Checksum Record

5.10.2.1.2 **3DSMV Pseudo-File Field Structure.** The field structure of each of these records shall be as described below.

5.10.2.1.2.1 **3DSMV Identifier Record.** This record shall consist of the ASCII string '3DSMV'.

5.10.2.1.2.2 **File Name Record.** This record shall consist of the ASCII string 'ssGnnnnnn3DS.VTX', where "ss" is the security code, and "nnnnnn" is the GTDB identifier.

5.10.2.1.2.3 **Vertex Record.** The field structure of this record shall be as follows:

    Coordinate Field

5.10.2.1.2.4 **Pseudo-EOF Record.** This record shall consist of the ASCII string 'EOF MDL3DSnnnnn.VTX', where 'nnnnn' is the number of the model described by this pseudo-file.

5.10.2.1.2.5 **Checksum Record.** The field structure of this record is as follows:

    Checksum Field

5.11 **3-D Dynamic Model (3DDM) Library File.** There may be one 3-D Dynamic Model (3DDM) Library File following the 3DSMV. The 3-D Dynamic Model Library File is optional in a GTDB.

5.11.1 **3DDM Record Order.** When included, the record order of the 3DDM file shall be as follows:

3DDM Identifier Record
File Name Record
3DDM (Model Library) Header Record
for each model
  Model Header Record
  for each model LOD
    LOD Header Record
    LOD Texture Reference Pointer Record(s) [optional]
    for each component
      Component Header Record
      Component Texture Reference Pointer Record(s) [optional]
      for each model polygon
        Model Polygon Record
        Microdescriptor Record(s) [optional]
        Vertex Pointer Records
        Polygon FACS Record(s) [optional]
        Polygon Texture Reference Pointer Record(s) [optional]
    Subsidiary Model References Record(s) [optional]
    for each point light string
      Point Light String Record(s) [optional]
      Point Light String FACS Record(s) [optional]
    Model FACS Record(s) [optional]
    Face-Based Texture Reference Record(s) [optional]
    Vertex-to-Vertex Texture Reference Record(s) [optional]
    Model-Based Texture Reference Record(s) [optional]
    Non-Mapped Texture Reference Record(s) [optional]
    Separation Plane Record(s) [optional]
    Collision Test Point Record(s) [optional]
Checksum Record

5.11.2 **3DDM Field Structure.** The field structure of each of these records shall be as described below.

5.11.2.1 **3DDM Identifier Record.** This record shall consist of the ASCII string '3DDM'.

5.11.2.2 **File Name Record.** This record shall consist of the ASCII string 'ssGnnnnnn3DD.LIB', where "ss" is the security code, and "nnnnnn" is the GTDB identifier.

5.11.2.3 **3DDM Header Record.** The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
Model Library Type Field
Security Level Field
Last Update Date Field
Number of Models Field

5.11.2.4 **Model Header Record.** The number of these records shall correspond to the value contained in the Number of Models field in the parent 3DDM Header record. The field structure of this record shall be as follows:

Model Number Field
Model Name Field
Model Description Field
Generic Model Flag Field
Feature Descriptor Code Field
Number of Model LODs Field

5.11.2.5 **LOD Header Record.** The number of these records for a given model group shall correspond to the value contained in the Number of Model LODs field in the Parent Model Header record. The field structure of this record shall be as follows:

Model Number Field
Model LOD Field
LOD Resolution Description Field
Sensor Types Supported Field
Source Simulator Field
Directivity Field
Radius Field
Predominant Height Field
Centroid Field
Base Polygon ID Field
Percentage of Texture Coverage Field
Number of Polygons Field
Number of LOD Texture Reference Pointers Field
Number of Components Field
Number of Subsidiary Model References Field
Number of Point Light Strings Field
Number of Model FACS Field
Number of Face-Based Texture References Field
Number of Vertex-to-Vertex Texture References Field
Number of Model-Based Texture References Field
Number of Non-Mapped Texture References Field
Number of Separation Planes Field
Number of Collision Test Points Field
Placement Point Field [optional]

5.11.2.6 **LOD Texture Reference Pointer Record.** The number of these records for a given model LOD shall correspond to the value contained in the Number of LOD Texture Reference Pointers field in the parent LOD Header record. The field structure of this record shall be as follows:

Texture Mapping Type Field
Texture Reference Table Index Field
Texture Mapping Set ID Field

5.11.2.7 **Component Header Record.** The number of these records for a given model LOD shall correspond to the value contained in the **Number** of Components field in the parent LOD Header record. The field structure of this record shall be as follows:

Component ID Field
Number of Components Texture Reference Pointers Field
Number of Polygon Field

5.11.2.8 **Component Texture Reference Pointer Record.** The number of these records for a given model LOD shall correspond to the value contained in the Number of Component Texture Reference Pointers field in the parent Component Header record. The field structure of this record shall be as follows:

Texture Mapping Type Field
Texture Reference Table Index Field
Texture Mapping Set ID Field

5.11.2.9 **Model Polygon Record.** The number of these records for a given component shall correspond to the value contained in the Number of Polygons field in the parent Component Header record. The last polygon shall define the model footprint, referenced in the Base Polygon ID Field of the Parent Model Record. The field structure of this record shall be as follows:

Polygon ID Field
Cluster ID Field
Component ID Field
Surface Material Category Field
Surface Material Subtype Field
Reflectance Field
Light Type Field
Specular Field
Polygon Non-Shadow Field
Polygon Normal Field
Transmissivity Field
Polygon Long Dimension Field
Polygon Short Dimension Field
Centroid Field
Diffuse Reflectance Field
Feature Onset Field
Layer Number (Radar) Field
Color Characteristics Field
Shading Type Field
Translucency Field
Polygon Non-Occulting Field
Cycle Rate Off Field
Cycle Rate On Field
Cycle Rate Field
Directionality Field
Light Horizontal Width Field
Light Horizontal Center Field
Light Horizontal Fall Field
Light Intensity Field
Light Vertical Width Field
Light Vertical Center Field
Light Vertical Fall Field
Polygon Illumination Type Field
Polygon Landing Light Illumination Field
Absorptivity Field
Emissivity Field
Exitance Field
Self-Emitter Field
Layer Number (Visual) Field
Layer Number (Infrared) Field
Number of Microdescriptors Field
Number of Vertices Field (Always three or greater)
Number of Polygon Texture Reference Pointers Field
Number of Polygon FACS Field

5.11.2.10 **Model Microdescriptor Record.** The number of these records shall correspond to the value in the Number of Microdescriptors field in the parent Model Polygon record. The field structure of each record shall be as follows:

Microdescriptor Type Field
Microdescriptor Value Field

5.11.2.11 **Vertex Pointer Record.** The number of records shall correspond to the Number of Vertices field within the parent Model Polygon record. Polygons shall be closed implicity, i.e., the first vertex shall not be repeated as the last. The Normal List Position Field shall be zero if vertex normals have not been requested. The field structure of each record is as follows:

Vertex List Position Field
Normal List Position Field [Always Zero]
Correlation Priority Field

5.11.2.12 **Polygon FACS Record.** The number of these records shall correspond to the value in the Number of Polygon FACS field in the parent Model Polygon record. The field structure of each record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Length of Attribute Field
Attribute Value Field

5.11.2.13 **Polygon Texture Reference Pointer Record.** The number of these records for a given model polygon shall correspond to the value of the Number of Polygon Texture Reference Pointer field within the parent Model Polygon record. The field structure of each record shall be as follows:

Texture Mapping Type Field
Texture Reference Table Index Field
Texture Mapping Set ID Field

5.11.2.14 **Subsidiary Model Reference Record.** The number of these records for a given model shall correspond to the value contained in the Number of Subsidiary Model References field in the parent Model record. The field structure of this record shall be as follows:

Referenced Model Library Type Field
Referenced Model Number Field
Referenced Model LOD Field
Translation Field
Scale Factor Field
Rotation Angles Field
Articulated Part Flag Field

5.11.2.15 Point Light String Record. The number of Point Light String records shall correspond to the value in the Number of Point Light Strings field within the LOD Header record. The field structure of this record shall be as follows:

Length Field
Orientation Field
Shape Code Field
Width Field
Directionality Field
Light Type Field
Predominant Height Field
Surface Material Category Field
Color Characteristics Field
Layer Number Field
Absorptivity Field
Centroid Field
Cycle Rate Off Time Field
Cycle Rate On Time Field
Diffuse Reflectance Field
Directivity (Infrared) Field
Directivity (Radar) Field
Directivity Field
Emissivity Field
Exitance Field
Feature Onset Field
Internal Material Category Field
Internal Material Volume Field
Layer Number (Infrared) Field
Light Horizontal Center Field
Light Horizontal Fall Field
Light Horizontal Width Field
Light Intensity Field
Light Vertical Center Field
Light Vertical Fall Field
Light Vertical Width Field
Long Lineal Field
Low Level Effects Field
Object Volume Field
Radius Field
Reflectance Field
Self-Emitter Field
Surface Material Subtype Field
Texture Map Reflectance Field
Transmissivity Field
Visible Range Field
Number of FACS Table Entries Field
Number of Lights Field
for each light in the string
    Position Field

5.11.2.16 **Point Light String FACS Record.** The number of these records shall correspond to the value in the Number of FACS Table Entries field in the parent Point Light String Record. The field structure of each record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Length of Attribute Field
Attribute Value Field

5.11.2.17 **Model FACS Record.** The number of these records shall correspond to the value in the Number of Model FACS field in the parent Model record. The field structure of each record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Length of Attribute Field
Attribute Value Field

5.11.2.18 **Face-Based Texture Reference Record.** The number of these records shall correspond to the value of the Number of Face-Based Texture References field in the parent LOD Header Record. The field structure of this record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific Or Generic Texture Flag Field
Texture Origin Field
Boundary ID Field
Mirror Field
Wrap Field
Wrap Type Field
Texture Scale Field
Polygon Alignment Vector Field
Rotation About Texture Origin Field
Polygon Reference Point Field
Layer Number Field

5.11.2.19 **Vertex-to-Vertex Texture Reference Record.** The number of these records shall correspond to the value of the Number of Vertex-to-Vertex Texture References field in the parent LOD Header Record. The field structure of this record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific Or Generic Texture Flag Field
Layer Number Field
Number of Texture Pattern Coordinates Field
for each texture pattern vertex
    Texture Pattern Coordinates Field

5.11.2.20 **Model-Based Texture Reference Record.** The number of these records shall correspond to the value of the Number of Model-Based Texture References field in the parent LOD Header Record. The field structure of this record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific Or Generic Texture Flag Field
Texture Origin Field
Boundary ID Field
Mirror Field
Wrap Field
Wrap Type Field
Texture Scale Field
Orientation Vectors Field
Model Reference Point Field
Layer Number Field

5.11.2.21 **Non-Mapped Texture Reference Record.** The number of these records shall correspond to the value contained in the Number of Non-Mapped Texture References field in the parent LOD Header record. The field structure of this record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific Or Generic Texture Flag Field

5.11.2.22 **Separation Plane Record.** The number of these records for a given model LOD shall correspond to the value contained in the Number of Separation Planes field in the parent LOD record. The field structure of this record shall be as follows:

Polygon ID Field
Separation Plane Number Field
Separation Plane Coefficients Field

5.11.2.23 **Collision Test Point Record.** The number of these records shall correspond to the value in the Number of Collision Test Points field within the parent LOD Header record. The field structure of each record shall be as follows:

Vertex List Position Field

5.11.2.24 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.12 **3-D Dynamic Model Vertex (3DDMV) File.** Following the 3DDM file, there shall be one 3-D Dynamic Model Vertex File. The 3DDMV shall be required if there is a 3DDM, but shall be omitted otherwise.

5.12.1 **3DDMV File Structure.** There shall be one pseudo-file for each model, containing the vertices used to define all the LODs of that model. Each pseudo-file shall be terminated by a special record indicating a pseudo-EOF (end of file). The pseudo-file structure within this file is as follows:

for each model in the 3DDM
  3DDMV Pseudo-File

5.12.2 **3DDMV Record Structure.** The record structure of each 3DDMV pseudo-file shall be as defined in the following subsections.

5.12.2.1 **3-D Dynamic Model Vertex Pseudo-Files.** There shall be one pseudo-file for each model, containing the vertices used to define all the LODs of that model. These pseudo-files shall physically occur on the tape in the same sequence as their corresponding model definitions occur within the model library.

5.12.2.1.1 **3DDMV Pseudo-File Record Order.** The record order of each 3DDMV pseudo-file shall be as follows:

3DDMV Identifier Record
File Name Record
for each vertex, vertex normal, and collision test point in 3DDM
  Vertex Record
Pseudo-EOF Record
Checksum Record

5.12.2.1.2 **3DDMV Pseudo-File Field Structure.** The field structure of each of these records shall be as described below.

5.12.2.1.2.1 **3DDMV Identifier Record.** This record shall consist of the ASCII string '3DDMV'.

5.12.2.1.2.2 **File Name Record.** This record shall consist of the ASCII string 'ssGnnnnnn3DD.VTX', where "ss" is the security code, and "nnnnnn" is the GTDB identifier.

5.12.2.1.2.3 **Vertex Record.** The field structure of this record shall be as follows:

Coordinate Field

5.12.2.1.2.4 **Pseudo-EOF Record.** This record shall consist of the ASCII string 'EOF MDL3DDnnnnn.VTX', where 'nnnnn' is the number of the model described by this pseudo-file.

5.12.2.1.2.5 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.13 Simulator Level of Detail Area Blocks File (SLAB). There shall be one Simulator Level of Detail Area Blocks File (SLAB) for each Simulator Level of Detail (SLOD) in the GTDB. The number of SLABs shall correspond to the value contained in the Simulator Level of Detail Count Field in the GTDB Parameters Record of the Gaming Area Header File. The filename for each SLAB shall take the form "ssGnnnnnnVmmmmmm.dd", where "ss" is the security code, "nnnnnn" is the GTDB identifier, "mmmmmm" is the version number, and "dd" is the SLOD number.

5.13.1 **SLAB File Structure.** For each area block, there shall be one pseudo-file for each type of data needed to describe the contents of that area block. The pseudo-file structure of each SLAB shall be as follows:

```
for each area block in SLOD
   Vertex Area Block (VAB) Pseudo-File
   Areal Feature Area Block (AFAB) Pseudo-File
   Linear Feature Area Block (LFAB) Pseudo-File [optional]
   Point Feature Area Block (PFAB) Pseudo-File [optional]
   Point Light Feature Area Block (PLFAB) Pseudo-File [optional]
   Point Light String Feature Area Block (PLSFAB) Pseudo-File [opt.]
   Terrain Polygon Area Block (TPAB) Pseudo-File [optional]
   Terrain Grid Area Block (TGAB) Pseudo-File [optional]
   Model Reference Area Block (MRAB) Pseudo-File [optional]
   Area Block Pseudo-EOF Record
```

5.13.2 **SLAB Pseudo-File Record Structure.** The record structure of each of these pseudo-files shall be as described in the following subsections.

5.13.2.1 **Vertex Area Block (VAB) Pseudo-File.** For every area block in the data base (previously identified in the ABH), there shall be one Vertex Area Block Pseudo-File associated with it.

5.13.2.1.1 **VAB Pseudo-File Record Order.** The record order of the vertex area block pseudo-file shall be as follows:

```
VAB Identifier Record
File Name Record
Vertex Area Block Header Record
for each culture/terrain vertex and vertex normal in area block
   Vertex Record
Pseudo-EOF Record
Checksum Record
```

5.13.2.1.2 **VAB Pseudo-File Field Structure.** The field structure of each of these records shall be as described below.

5.13.2.1.2.1 **VAB Identifier Record.** This record shall consist of the ASCII string 'VAB'.

5.13.2.1.2.2 **File Name Record.** This record shall consist of the ASCII string "VABnnnnnnnnnn.ss", where "nnnnnnnnnn" is the area block number and "ss" is the SLOD number.

5.13.2.1.2.3 **Vertex Area Block Header Record.** The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
Area Block ID Field
Lat/Long SW Corner Field
Lat/Long NE Corner Field
Last Update Date Field
Security Level Field
Number of Vertices Field (Always four or greater)
End Vertex ID Field

5.13.2.1.2.4 **Vertex Record.** The field structure of this record shall be as follows:

Vertex List Position Field
Coordinate Field

5.13.2.1.2.5 **Pseudo-EOF Record.** This record shall consist of the ASCII string 'EOF VABnnnnnnnnnn.ss', where 'nnnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.1.2.6 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.13.2.2 **Areal Feature Area Block (AFAB) Pseudo-File.** For each area block in the data base, there shall be one Areal Feature Area Block Pseudo-File associated with it.

5.13.2.2.1 **AFAB Pseudo-File Record Order.** The record order of the AFAB pseudo-file shall be as follows:

AFAB Identifier Record
File Name Record
Feature Area Block Header Record
Face-Based Texture Reference Record(s) [optional]
Global-Based Texture Reference Record(s) [optional]
for each feature
  Areal Feature Record
  Microdescriptor Record(s) [optional]
  FACS Record(s) [optional]
  Vertex Pointer Records
  Non-Mapped Texture Reference Record(s) [optional]
  Mapped Texture Reference Pointer Record(s) [optional]
Pseudo-EOF Record
Checksum Record

5.13.2.2.2 **AFAB Pseudo-File Field Structure.** The field structure of each of these records shall be as described below.

5.13.2.2.2.1 **AFAB Identifier Record.** This record shall consist of the ASCII string 'AFAB'.

5.13.2.2.2.2 **File Name Record.** This record shall consist of the ASCII string 'AFABnnnnnnnnnn.ss', where 'nnnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.2.2.3 **Feature Area Block Header Record.** The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
Area Block ID Field
Lat/Long SW Corner Field
Lat/Long NE Corner Field
Last Update Date Field
Security Level Field
Number of Features Field
Number of Face-Based Texture References Field
Number of Global-Based Texture References Field

5.13.2.2.2.4 **Face-Based Texture Reference Record.** The number of these records shall correspond to the value of the Number of Face-Based Texture References field in the parent Feature Area Block Header Record. The field structure of this record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific Or Generic Texture Flag Field
Texture Origin Field
Boundary ID Field
Mirror Field
Wrap Field
Wrap Type Field
Texture Scale Field
Polygon Alignment Vector Field
Rotation About Texture Origin Field
Polygon Reference Point Field
Layer Number Field

5.13.2.2.2.5 **Global-Based Texture Reference Record.** The number of these records shall correspond to the value of the Number of Global-Based Texture References field in the parent Feature Area Block Header Record. The field structure of this record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific Or Generic Texture Flag Field
Texture Origin Field
Boundary ID Field
Mirror Field
Wrap Field
Wrap Type Field
Texture Scale Field
Orientation Vectors Field
Global Reference Point Field
Layer Number Field

5.13.2.2.2.6 **Areal Feature Record.** The total number of these records shall correspond to the value contained in the Number of Features field in the parent Feature Area Block Header record. The field structure of this record shall be as follows:

Feature Number Field
FID Code Field
Feature Descriptor Code
Synthetic Data Flag Field
Correlation Priority Field
Reflectance Field
Predominant Height Field
Feature Fragment Flag Field
Superfeature Number Field
Surface Material Category Field
Surface Material Subtype Field
Specular Field
Number of Structures Field
Percent of Roof Coverage Field
Roof Type Field
Monitor Type Field
Polygon Normal Field
Percent of Tree Coverage Field
Shape Code Field
Centroid Field
Directivity (Radar) Field
Layer Number (Radar) Field
Diffuse Reflectance Field
Feature Onset Field
Low Level Effects Field
Absorptivity Field
Directivity (Visual) Field
Directivity (Infrared) Field
Emissivity Field
Exitance Field
Transmissivity Field
Color Characteristics Field
Self-Emitter Field
Translucency Field
Shading Type Field
Layer Number (Visual) Field
Layer Number (Infrared) Field
Radius Field
Blending Type Field
Polygon Illumination Type Field
Number of Microdescriptors Field
Number of Non-Mapped Texture References Field
Number of Mapped Texture Reference Pointers Field
Number of Culture Vertices Field (Always three or greater)
Number of FACS Field

5.13.2.2.2.7 **Microdescriptor Record.** The number of these records shall correspond to the value in the Number of Microdescriptors field in the parent Areal Feature record. The field structure of each record shall be as follows:

Microdescriptor Type Field
Microdescriptor Value(s) Field

5.13.2.2.2.8 **FACS Record.** The number of these records shall correspond to the value in the Number of FACS field in the parent Areal Feature record. The field structure of each record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Length of Attribute Field
Attribute Value Field

5.13.2.2.2.9 **Vertex Pointer Record.** The number of records shall correspond to the Number of Culture Vertices field within the parent Areal Feature record. Areal features shall be closed implicitly, i.e., the first vertex shall not be repeated as the last. Vertices shall be ordered in a counterclockwise direction, as viewed from above. The field structure of each record shall be as follows:

Vertex List Position Field
Correlation Priority Field

5.13.2.2.2.10 **Non-Mapped Texture Reference Record.** The number of these records shall correspond to the value of the Number of Non-Mapped Texture References field within the parent Areal Feature record. The field structure of each record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific or Generic Texture Flag Field

5.13.2.2.2.11 **Mapped Texture Reference Pointer Record.** The number of these records shall correspond to the value of the Number of Mapped Texture Reference Pointers field within the parent Areal Feature record. The field structure of each record shall be as follows:

Texture Mapping Type Field
Texture Reference Table Index Field
Texture Mapping Set ID Field

5.13.2.2.2.12 **Pseudo-EOF Record.** This record shall consist of the ASCII string 'EOF AFABnnnnnnnnnn.ss', where 'nnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.2.2.13 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.13.2.3 **Linear Feature Area Block (LFAB) Pseudo-File.** For each area block which includes linear features, there shall be one Linear Feature Area Block Pseudo-File associated with it. The LFAB shall be included when an area block contains at least one linear feature.

5.13.2.3.1 **LFAB Pseudo-File Record Order.** The record order of the LFAB pseudo-file shall be as follows:

LFAB Identifier Record
File Name Record
Feature Area Block Header Record
for each feature
  Linear Feature Record
  Microdescriptor Record(s) [optional]
  FACS Record(s) [optional]
  Vertex Pointer Records
  Non-Mapped Texture Reference Record(s) [optional]
Pseudo-EOF Record
Checksum Record

5.13.2.3.2 **LFAB Pseudo-File Field Structure.** The field structure of each of these records shall be as described below.

5.13.2.3.2.1 **LFAB Identifier Record.** This record shall consist of the ASCII string 'LFAB'.

5.13.2.3.2.2 **File Name Record.** This record shall consist of the ASCII string 'LFABnnnnnnnnnn.ss', where 'nnnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.3.2.3 **Feature Area Block Header Record.** This record shall contain control information describing the file. The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
Area Block ID Field
Lat/Long SW Corner Field
Lat/Long NE Corner Field
Last Update Date Field
Security Level Field
Number of Features Field

5.13.2.3.2.4 **Linear Feature Record.** The total number of these records shall correspond to the value contained in the Number of Features field in the parent Feature Area Block Header record. The field structure of this record shall be as follows:

Feature Number Field
FID Code Field
Feature Descriptor Code Field
Synthetic Data Flag Field
Correlation Priority Field
Reflectance Field
Predominant Height Field
Feature Fragment Flag Field
Superfeature Number Field
Surface Material Category Field
Surface Material Subtype Field
Specular Field
Width Field
Diffuse Reflectance Field
Directivity (Radar) Field
Layer Number (Radar) Field
Feature Onset Field
Low Level Effects Field
Directivity (Visual) Field
Directivity (Infrared) Field
Layer Number (Visual) Field
Layer Number (Infrared) Field
Absorptivity Field
Emissivity Field
Exitance Field
Color Characteristics Field
Self-Emitter Field
Translucency Field
Radius Field
Blending Type Field
Centroid Field
Transmissivity Field
Number of Microdescriptors Field
Number of Non-Mapped Texture References Field
Number of Culture Vertices Field (Always two or greater)
Number of FACS Field

5.13.2.3.2.5 **Microdescriptor Record.** The number of these records shall correspond to the value in the Number of Microdescriptors field in the parent Linear Feature record. The field structure of each record shall be as follows:

Microdescriptor Type Field
Microdescriptor Value Field

5.13.2.3.2.6 **FACS Record.** The number of these records shall correspond to the value in the Number of FACS field in the parent Linear Feature record. The field structure of each record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Length of Attribute Field
Attribute Value Field

5.13.2.3.2.7 **Vertex Pointer Record.** The number of records shall correspond to the Number of Culture Vertices field within the parent Linear Feature record. The field structure of each record shall be as follows:

Vertex List Position Field
Correlation Priority Field

5.13.2.3.2.8 **Non-Mapped Texture Reference Record.** The number of these records shall correspond to the value of the Number of Photo Texture References field within the parent Linear Feature record. The field structure of each record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific or Generic Texture Flag Field

5.13.2.3.2.9 **Pseudo-EOF Record.** This record shall consist of the ASCII string 'EOF LFABnnnnnnnnnn.ss', where 'nnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.3.2.10 **Checksum Record.** The field structure of this record is as follows:

Checksum Field

5.13.2.4 **Point Feature Area Block (PFAB) Pseudo-File.** For each area block which includes point features, there shall be one Point Feature Area Block Pseudo-File associated with it. The PFAB shall be included when an area block contains at least one point feature.

5.13.2.4.1 **PFAB Pseudo-File Record Order.** The record order of the PFAB pseudo-file shall be as follows:

PFAB Identifier Record
File Name Record
Feature Area Block Header Record
for each feature
  Point Feature Record
  Microdescriptor Record(s) [optional]
  FACS Record(s) [optional]
  Vertex Pointer Record(s)
  Non-Mapped Texture Reference Record(s) [optional]
Pseudo-EOF Record
Checksum Record

5.13.2.4.2 **PFAB Pseudo-File Field Structure.** The field structure of each of these records shall be as described below.

5.13.2.4.2.1 **PFAB Identifier Record.** This record shall consist of the ASCII string 'PFAB'.

5.13.2.4.2.2 **File Name Record.** This record shall consist of the ASCII string 'PFABnnnnnnnnnn.ss', where 'nnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.4.2.3 **Feature Area Block Header Record.** The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
Area Block ID Field
Lat/Long SW Corner Field
Lat/Long NE Corner Field
Last Update Date Field
Security Level Field
Number of Features Field

5.13.2.4.2.4 **Point Feature Record.** The total number of these records shall correspond to the value contained in the Number of Features field in the parent Feature Area Block Header record. The field structure of this record shall be as follows:

Feature Number Field
FID Code Field
Feature Descriptor Code Field
Synthetic Data Flag Field
Correlation Priority Field
Reflectance Field
Predominant Height Field
Feature Fragment Flag Field
Superfeature Number Field
Surface Material Category Field
Surface Material Subtype Field
Specular Field
Length Field
Width Field
Radius Field
Orientation Field
Shape Code Field
Directivity (Radar) Field
Diffuse Reflectance Field
Feature Onset Field
Low Level Effects Field
Long Lineal Field
Directivity (Visual) Field
Directivity (Infrared) Field
Absorptivity Field
Emissivity Field
Exitance Field
Transmissivity Field
Color Characteristics Field
Self-Emitter Field
Translucency Field
Blending Type Field
Centroid Field
Number of Microdescriptors Field
Number of Non-Mapped Texture References Field
Number of Culture Vertices Field
Number of FACS Field

5.13.2.4.2.5 **Microdescriptor Record.** The number of these records shall correspond to the value in the Number of Microdescriptors field in the parent Point Feature record. The field structure of each record shall be as follows:

Microdescriptor Type Field
Microdescriptor Value Field

5.13.2.4.2.6 **FACS Record.** The number of these records shall correspond to the value in the Number of FACS field in the parent Point Feature record. The field structure of each record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Length of Attribute Field
Attribute Value Field

5.13.2.4.2.7 **Vertex Pointer Record.** The number of records shall correspond to the Number of Vertices field within the parent Point Feature record. The field structure of each record shall be as follows:

Vertex List Position Field
Correlation Priority Field

5.13.2.4.2.8 **Non-Mapped Texture Reference Record.** The number of these records shall correspond to the value of the Number of Photo Texture References field within the parent Point Feature record. The field structure of each record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture ID Field
Specific or Generic Texture Flag Field

5.13.2.4.2.9 **Pseudo-EOF Record.** This record shall consist of the ASCII string 'EOF PFABnnnnnnnnnn.ss', where 'nnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.4.2.10 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.13.2.5 **Point Light Feature Area Block (PLFAB) Pseudo-File.** For each area block which includes point light features, there shall be one Point Light Feature Area Block Pseudo-File associated with it. The PLFAB shall be included when an area block contains at least one point light feature.

5.13.2.5.1 **PLFAB Pseudo-File Record Order.** The record order of the PLFAB pseudo-file shall be as follows:

PLFAB Identifier Record
File Name Record
Feature Area Block Header Record
for each feature
  Point Light Feature Record
  Microdescriptor Record(s) [optional]
  FACS Record(s) [optional]
  Vertex Pointer Record
Pseudo-EOF Record
Checksum Record

5.13.2.5.2 **PLFAB Pseudo-File Field Structure.** The field structure of each of these records shall be as described below.

5.13.2.5.2.1 **PLFAB Identifier Record.** This record shall consist of the ASCII string 'PLFAB'.

5.13.2.5.2.2 **File Name Record.** This record shall consist of the ASCII string 'PLABnnnnnnnnnn.ss', where 'nnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.5.2.3 **Feature Area Block Header Record.** This record shall contain control information describing the file. The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
Area Block ID Field
Lat/Long SW Corner Field
Lat/Long NE Corner Field
Last Update Date Field
Security Level Field
Number of Features Field

5.13.2.5.2.4 **Point Light Feature Record.** This record shall contain control fields and attributes describing a particular point light feature. The total number of these records shall correspond to the value contained in the Number of Features field in the parent Feature Area Block Header record. The field structure of this record shall be as follows:

Feature Number Field
FID Code Field
Feature Descriptor Code Field
Synthetic Data Flag Field
Correlation Priority Field
Absorptivity Field
Directivity (Infrared) Field
Emissivity Field
Exitance Field
Reflectance Field
Transmissivity Field
Color Characteristics Field
Predominant Height Field
Surface Material Category Field
Directivity (Visual) Field
Self-Emitter Field
Directionality Field
Cycle Rate Off Field
Cycle Rate On Field
Light Horizontal Center Field
Light Horizontal Fall Field
Light Horizontal Width Field
Light Intensity Field
Light Type Field
Light Vertical Width Field
Light Vertical Center Field
Light Vertical Fall Field
Length Field
Width Field
Radius Field
Blending Type Field
Centroid Field
Orientation Field
Shape Code Field
Layer Number (Visual) Field
Layer Number (Infrared) Field
Visible Range Field
Number of Microdescriptors Field
Number of Texture References Field [N/A]
Number of Culture Vertices Field
Number of FACS Field

5.13.2.5.2.5 **Microdescriptor Record.** The number of these records shall correspond to the value in the Number of Microdescriptors field in the parent Point Light Feature record. The field structure of each record shall be as follows:

Microdescriptor Type Field
Microdescriptor Value Field

5.13.2.5.2.6 **FACS Record.** The number of these records shall correspond to the value in the Number of FACS field in the parent Point Light Feature record. The field structure of each record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Length of Attribute Field
Attribute Value Field

5.13.2.5.2.7 **Vertex Pointer Record.** There shall be exactly one of these records defining the location of the point light feature. The value '1' shall be stored in the Number of Vertices field within the parent Point Light Feature record. The field structure of this record shall be as follows:

Vertex List Position Field
Correlation Priority Field

5.13.2.5.2.8 **Pseudo-EOF Record.** This record shall consist of the ASCII string 'EOF PLABnnnnnnnnnn.ss', where 'nnnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.5.2.9 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.13.2.6 **Point Light String Feature Area Block (PLSFAB) Pseudo-File.** For each area block which includes point light strings, there shall be one Point Light String Feature Area Block Pseudo-File associated with it. The PLSFAB shall be included when an area block contains at least one point light string feature.

5.13.2.6.1 **PLSFAB Pseudo-File Record Order.** The record order of the PLSFAB pseudo-file shall be as follows:

PLSFAB Identifier Record
File Name Record
Feature Area Block Header Record
for each feature
   Point Light String Feature Record
   Microdescriptor Record(s) [optional]
   FACS Record(s) [optional]
   Vertex Pointer Record(s)
Pseudo-EOF Record
Checksum Record

5.13.2.6.2 **PLSFAB Pseudo-File Field Structure.** The field structure of each of these records shall be as described below.

5.13.2.6.2.1 **PLSFAB Identifier Record.** This record shall consist of the ASCII string 'PLSFAB'.

5.13.2.6.2.2 **File Name Record.** This record shall consist of the ASCII string 'PSABnnnnnnnnnn.ss', where 'nnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.6.2.3 **Feature Area Block Header Record.** The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
Area Block ID Field
Lat/Long SW Corner Field
Lat/Long NE Corner Field
Last Update Date Field
Security Level Field
Number of Features Field

5.13.2.6.2.4 **Point Light String Feature Record.** The total number of these records shall correspond to the value contained in the Number of Features field in the parent Feature Area Block Header record. The field structure of this record shall be as follows:

Feature Number Field
FID Code Field
Feature Descriptor Code Field
Synthetic Data Flag Field
Correlation Priority Field
Absorptivity Field
Directivity (Infrared) Field
Emissivity Field
Exitance Field
Reflectance Field
Transmissivity Field
Color Characteristics Field
Predominant Height Field
Surface Material Category Field
Directivity (Visual) Field
Self-Emitter Field
Feature Fragment Flag Field
Superfeature Number Field
Light String Shape Field
Number of Lights Field
Directionality Field
Cycle Rate Off Time Field
Cycle Rate On Time Field
Light Horizontal Center Field
Light Horizontal Fall Field
Light Horizontal Width Field
Orientation Field
Light Intensity Field
Light Type Field
Light Vertical Width Field
Light Vertical Center Field
Light Vertical Fall Field
Blending Type Field
Centroid Field
Radius Field
Length Field
Width Field
Layer Number (Visual) Field
Layer Number (Infrared) Field
Light Delta Field
Visible Range Field
Number of Microdescriptors Field
Number of Texture References Field [N/A]
Number of Culture Vertices Field
Number of FACS Field

5.13.2.6.2.5 **Microdescriptor Record.** The number of these records shall correspond to the value in the Number of Microdescriptors field in the parent Point Light String Feature record. The field structure of each record shall be as follows:

Microdescriptor Type Field
Microdescriptor Value Field

5.13.2.6.2.6 **FACS Record.** The number of these records shall correspond to the value in the Number of FACS field in the parent Point Light String Feature record. The field structure of each record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Length of Attribute Field
Attribute Value Field

5.13.2.6.2.7 **Vertex Pointer Record.** The number of Vertex Pointer records shall correspond to the Number of Vertices field within the parent Point Light String Feature record. The first record shall define the origin of the Point Light String. When the Light String Shape Field of the parent Point Light String Feature Record indicates that the lights fall within a straight line, there shall be no vertex pointer records other than the first. The field structure of each record shall be as follows:

Vertex List Position Field
Correlation Priority Field

5.13.2.6.2.8 **Pseudo-EOF Record.** This record shall consist of the ASCII string 'EOF PSABnnnnnnnnnn.ss', where 'nnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.6.2.9 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.13.2.7 **Terrain Polygon Area Block (TPAB) Pseudo-File.** For a GTDB which includes polygonized terrain, there shall be one Terrain Polygon Area Block Pseudo-File associated with each area block. The TPAB shall be included when polygonized terrain has been requested.

5.13.2.7.1 **TPAB Pseudo-File Record Order.** The record order of the TPAB pseudo-file shall be as follows:

TPAB Identifier Record
File Name Record
Terrain Area Block Beader Record
for each terrain polygon
    Terrain Polygon Record
    Vertex List Pointer Records
    Culture Reference Record(s) [optional]
    Vertex-to-Vertex Texture Reference Record(s) [optional]
Pseudo-EOF Record
Checksum Record

5.13.2.7.2 **TPAB Pseudo-File Field Structure.** The field structure of each of these records shall be as described below.

5.13.2.7.2.1 **TPAB Identifier Record.** This record shall consist of the ASCII string 'TPAB'.

5.13.2.7.2.2 **File Name Record.** This record shall consist of the ASCII string 'TPABnnnnnnnnnn.ss', where 'nnnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.7.2.3 **TPAB Header Record.** The field structure of this record shall be as described below:

Project 2851 GTDB Catalog ID Field
Area Block ID Field
Lat/Long SW Corner Field
Lat/Long NE Corner Field
Last Update Date Field
Security Level Field
Terrain Type Field
Latitude Interval Field
Longitude Interval Field
Number of Terrain Polygons Field
Column Count Field (Always Zero)
Row Count Field (Always Zero)

5.13.2.7.2.4 **Terrain Polygon Record.** The number of these records shall correspond to the value in the Number of Terrain Polygons field within the parent TPAB Header Record. The field structure of this record shall be as follows:

Terrain Polygon ID Field
Shape Code Field
Polygon Normal Field
Number of Culture References Field
Number of Vertices Field (Always three or greater)
Number of Vertex-to-Vertex Texture References Field

5.13.2.7.2.5 **Vertex List Pointer Record.** The actual number of records shall correspond to the Number of Vertices field within the parent Terrain Polygon record. Vertices shall be ordered in a counterclockwise direction, as viewed from above. All polygons shall be closed implicitly. The Normal List Position Field shall be zero when vertex normals have not been requested. The field structure of each record shall be as follows:

Vertex List Position Field
Normal List Position Field
Correlation Priority Field

5.13.2.7.2.6 **Culture Reference Record.** The number of these records shall correspond to the value of the Number of Culture References field within the parent Terrain Polygon record. The field structure of each record shall be as follows:

Type of Reference Field
Feature Type Field
Feature Number Field
Model Library Type Field
Model Reference Number Field
Feature Descriptor Code Field
Layer Number Field

5.13.2.7.2.7 **Vertex-to-Vertex Texture Reference Record.** There shall be one texture pattern vertex defined for each polygon vertex. The first texture pattern vertex shall map to the first polygon vertex. The number of these records shall correspond to the value of the Number of Vertex-to-Vertex Texture References field in the parent Terrain Polygon Record. The field structure of this record shall be as follows:

Texture Reference Table Index Field
GTDB Texture Library Type Field
Texture Mapping Set ID Field
Texture ID Field
Specific or Generic Texture Flag Field
Layer Number Field
Number of Texture Pattern Coordinates Field
for each texture pattern vertex
  Texture Pattern Coordinates Field

5.13.2.7.2.8 **Pseudo-EOF Record.** This record shall consist of the ASCII string 'EOF TPABnnnnnnnnnn.ss', where 'nnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.7.2.9 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.13.2.8 **Terrain Grid Area Block (TGAB) Pseudo-File.** For a GTDB which includes gridded terrain, there shall be one Terrain Grid Area Block Pseudo-File associated with every area block identified as being part of a SLOD. The TGAB shall be included when gridded terrain has been requested.

5.13.2.8.1 **TGAB Pseudo-File Record Order.** The record order of the TGAB pseudo-file shall be as follows:

TGAB Identifier Record
File Name Record
Terrain Area Block Header Record
for each terrain grid post
  Terrain Post Record
Pseudo-EOF Record
Checksum Record

5.13.2.8.2 **TGAB Pseudo-File Field Structure.** The field structure of each of these records shall be as described below.

5.13.2.8.2.1 **TGAB Identifier Record.** This record shall consist of the ASCII string 'TGAB'.

5.13.2.8.2.2 **File Name Record.** This record shall consist of the ASCII string 'TGABnnnnnnnnnn.ss', where 'nnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.8.2.3 **TGAB Header Record.** The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
Area Block ID Field
Lat/Long SW Corner Field
Lat/Long NE Corner Field
Last Update Date Field
Security Level Field
Terrain Type Field
Latitude Interval Field
Longitude Interval Field
Number of Terrain Polygons Field (Always Zero)
Column Count Field
Row Count Field

5.13.2.8.2.4 **Terrain Post Record.** The total number of these records shall be equal to the product of the values in the Column Count and Row Count fields within the TGAB Header Record. Elevation posts shall be sequenced from the southwest corner of the area block to the northeast, incrementing in latitude within each increment in longitude, at intervals specified in the parent TGAB Header Record. The field structure of each record shall be as follows:

Coordinate Field

5.13.2.8.2.5 **Pseudo-EOF Record.** This record shall consist of the ASCII string 'EOF TGABnnnnnnnnnn.ss', where 'nnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.8.2.6 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.13.2.9 **Model Reference Area Block (MRAB) Pseudo-File.** For every area block in which a model reference is present, there shall be one Model Reference Area Block Pseudo-File. The MRAB shall be included when there is at least one model reference within an area block.

5.13.2.9.1 **MRAB Pseudo-File Record Order.** The record order of the MRAB pseudo-file shall be as follows:

MRAB Identifier Record
File Name Record
Model Reference Area Block Header Record
for each model reference
   Model Reference Record
Pseudo-EOF Record
Checksum Record

5.13.2.9.2 **MRAB Pseudo-File Field Structure.** The field structure of each of these records shall be as described below.

5.13.2.9.2.1 **MRAB Identifier Record.** This record shall consist of the ASCII string 'MRAB'.

5.13.2.9.2.2 **File Name Record.** This record shall consist of the ASCII string 'MRABnnnnnnnnnn.ss', where 'nnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.9.2.3 **MRAB Header Record.** The field structure of this record shall be as follows:

Project 2851 GTDB Catalog ID Field
Area Block ID Field
Lat/Long SW Corner Field
Lat/Long NE Corner Field
Last Update Date Field
Security Level Field
Number of Model References Field

5.13.2.9.2.4 **Model Reference Record.** The number of these records shall correspond to the Number of Model Referenced Field in the MRAB header record. The field structure of this record shall be as follows:

Model Library Type Field
Model Number Field
Feature Number Field
Superfeature Number Field
Offset Vector Field
Orientation Angle Field
Rotation Field
Translation Field
Scale Factor Field
Correlation Priority Field
Terrain Polygon Overlap Flag Field

5.13.2.9.2.5 **Pseudo-EOF Record.** This record shall consist of the ASCII string 'EOF MRABnnnnnnnnnn.ss', where 'nnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.13.2.9.2.6 **Checksum Record.** The field structure of this record shall be as follows:

Checksum Field

5.13.2.10 **Area Block Pseudo-EOF Record.** This record shall consist of the ASCII string 'EOF ABnnnnnnnnnn.ss', where 'nnnnnnnnn' is the area block number and 'ss' is the SLOD number.

5.14 **Areal Texture (AT) Library.** The Areal Texture (AT) Library is optional in a GTDB. The AT shall consist of files in the National Imagery Transmission Format (NITF), or their native format. All NITF files shall follow the format described in the NITF Version 1.1 document dated 1 March 1989 and the NITF Change Notice Number 2 dated 23 May 1990.

5.14.1 **AT File Structure.** When included, the file order of the AT library shall be as follows:

```
NITF Header File
for each areal texture
   NITF Image Sub-Header File
   if texture is in Stage 1 then
      Original Format Image File(s) [optional]
   else
      NITF Image Data File [optional]
   end if
```

5.14.2 **AT File Record Structures.** The record structure of each of these files shall be as described below.

5.14.2.1 **NITF Header File.** The NITF Header File shall be included in every AT Library. The NITF Header File shall contain field labels on odd-numbered lines and field values (corresponding to the immediately preceding field label) on even-numbered lines. Each field label line shall be in all capital letters. Each line shall be terminated with an ASCII Carriage Return/Line Feed pair. The file shall be terminated with an ASCII end-of-file (CNTRL Z) character.

5.14.2.1.1 **Filename.** The ANSI filename for this file shall be 'ssGnnnnnnNITFi.HDR', where "ss" is the security code, "nnnnnn" is the GTDB identifier, and "i" is "A" for the areal texture library or "M" for the model texture library.

5.14.2.1.2 **File Format.** Valid data shall be provided in all fields for all texture types. The file format shall be as follows:

Label	Field
MHDR	Message Type & Version Field
STYPE	System Type Field
OSTAID	Originating Station ID Field
MDT	Message Date & Time Field
MTITLE	Message Title Field
MSCLAS	Message Security Classification Field
MSCODE	Message Codewords Field
MSCTLH	Message Control & Handling Field
MSREL	Message Releasing Instructions Field
MSCAUT	Message Classification Authority Field
MSCTLN	Message Security Control Number Field
MSDWNG	Message Security Downgrade Field
MSDEVT	Message Downgrading Event Field
MSCOP	Message Copy Number Field
MSCPYS	Message Number of Copies Field
ENCRYP	Encryption Field
ONAME	Originator's Name Field
OPHONE	Originator's Phone Number Field
ML	Message Length Field
HL	NITF Header Length Field
NUMI	Number of Images Field
LISH001	Length of 1st Image Sub-Header Field
LI001	Length of 1st Image Field
....	
LISHnnn	Length of Nth Image Sub-Header Field
LInnn	Length of Nth Image Field
NUMS	Number of Symbols Field [always 0]
NUML	Number of Labels Field [always 0]
NUMT	Number of Text Files Field [always 0]
NUMA	Number of Audio Segments Field [always 0]
NUMF	Number of Non-Static Presentations Field [always 0]
UDHDL	GTDB User Defined Header Data Length Field
	GTDB User Defined Header Data Fields
XHDL	Extended Header Data Length Field [always 0]
XHD	Extended Header Data Field [reserved]

5.14.2.1.2.1 **GTDB User Defined Header Data.** The GTDB User Defined Header Data format shall be as follows:

Label	Field
UDHD	Data Base Sentinel Field [always "GTDB"]

5.14.2.1.2.1.1 **Image Tie Point Data.** Valid data shall be provided in NUMGTP, GTPID, NUMGTPR, GTEXLIB, and GTEXID for Stage 1 and Stage 2 Areal Texture only. Valid data shall be provided in the remaining fields for model texture only. Zeroes shall be provided otherwise.

NUMGTP	Number of Geographic Tie Points Field
	for each geographic tie point
GTPID	Geographic Tie Point ID Field
NUMGTPR	Number of Tie Point References Field
	for each tie point reference
GTEXLIB	GTDB Texture Library Field
GTEXID	GTDB Texture ID Field
NUMMTP	Number of Model Tie Points Field
	for each model tie point
MTPID	Model Tie Point ID Field
NUMMTPR	Number of Tie Point References Field
	for each tie point reference
GTEXLIB	GTDB Texture Library Field
GTEXID	GTDB Texture ID Field

5.14.2.1.2.1.2 **Generic Texture Association Data.** Valid data shall be provided for Generic Texture. Zeroes shall be provided for all other texture types.

NUMGTS	Number of Generic Texture Sets Field
	for each generic texture set
GTSNAME	Generic Texture Set Name Field
OMTF	Object or Material Texture Flag Field
NUMGT	Number of Generic Textures In Set Field
	for each generic texture
GTEXID	GTDB Texture ID Field

5.14.2.2 **NITF Image Sub-Header File.** This file shall be included if areal texture is provided. The NITF Image Sub-Header File shall contain field labels on odd-numbered lines and field values (corresponding to the immediately preceding field label) on even-numbered lines. Each field label line shall be in all capital letters. Each line shall be terminated with an ASCII Carriage Return/Line Feed pair. The file shall be terminated with an ASCII end-of-file (CNTRL Z) character. Coordinate system names shall be spelled out. Image Geographic Location Coordinates shall specify the Southwest and Northeast corners of the image in thousandths of arc-seconds. Color data shall be stored directly in the NITF Image Data File. For SMC/FDC data, Look-up Table entries shall be 7-byte ASCII strings. All model textures shall be in a local cartesian coordinate system in units of meters. All generic textures shall be in units of meters, whether they be intended for models or geographic areas. All areal, or geographic, texture in Stages 1 and 2 shall be in the native source coordinate system. Areal textures in Stage 3 shall be in the geodetic coordinate system, while areal textures in Stages 4 and 5 shall be in the coordinate system as specified by the GTDB parameter set.

5.14.2.2.1 **Filename.** The ANSI filename for this file shall be 'ssGnnnnnnTEXitttt.HDR', where "ss" is the security code, "nnnnnn" is the GTDB identifier, "i" is "A" for the areal texture library or "M" for model texture library, and "ttttt" is the texture image number.

5.14.2.2.2 **File Format.** Valid data shall be provided in all fields for all texture types, with the exception of the ICORDS and IGEOLO fields. For these two fields, valid data shall be provided for all Areal, SMC/FDC, and Model Texture Types only. Zeroes shall be provided for all other types. The file format shall be as follows:

Label	Field
IM	Message Part Type Field
IID	Image (Texture) ID Field
IDATIM	Image Date & Time Field
TGTID	Target ID Field
ITITLE	Image Title (Texture Name) Field
ISCLAS	Image Security Classification Field
ISCODE	Image Codewords Field
ISCTLH	Image Control & Handling Field
ISREL	Image Releasing Instructions Field
ISCAUT	Image Classification Authority Field
ISCTLN	Image Security Control Number Field
ISDWNG	Image Security Downgrade Field
ISDEVT	Image Downgrading Event Field
ENCRYP	Encryption Field
ISORCE	Image Source Field
ICORDS	Image Coordinate System Field
IGEOLO	Image Geographic Location Field
NICOM	Number of Image Comments Field
	for each image comment
ICOMn	Image Comment Field n
IC	Image Compression Field
COMRAT	Compression Rate Code Field
NBANDS	Number of Bands Field
	for each band
ITYPEn	Image Type Field n
IFCn	Image Filter Condition Field n
IMFLTn	Standard Image Filter Code Field n
NLUTSn	Number of LUTs Field [always 0 or 1]
	for each LUT
NELUT1	Number of LUT Entries Field
	for each LUT entry e
LUTDe	LUT Entry Data
ISYNC	Image Sync Code Field
IMODE	Image Mode Field
NBPR	Number of Blocks Per Row Field
NBPC	Number of Blocks Per Column Field
NPPBH	Number of Pixels Per Block Horizontal Field
NPPBV	Number of Pixels Per Block Vertical Field
NBPP	Number of Bits Per Pixel Per Band Field
DLVL	Display Level Field [reserved]
ALVL	Attachment Field [reserved]
ILOC	Image Location Field [reserved]
IMAG	Image Magnification Field
UDIDL	GTDB User Defined Image Data Length Field
	GTDB User Defined Image Data Fields
XSHDL	Extended Sub-Header Data Length Field
XSHD	Extended Sub-Header Data Field [reserved]

5.14.2.2.2.1 **GTDB User Defined Image Data.** The GTDB User Defined Image Data format shall be as follows:

Label	Field
UDID	Data Base Sentinel Field [always "GTDB"]

5.14.2.2.2.1.1 **General Processing Data.** Valid data shall be provided for all texture types, with the following exceptions: MSTF, NRF, ORF, HRF, SMF, IICEF, ITICEF, 2GCF, 3GCF, ICAPDT, PAST, and GEOLOC shall contain valid data for all specific Areal and Model textures; MSTF, ICAPDT, PAST, and GEOLOC shall contain valid data for SMC/FDC texture; GTSNAME shall contain valid data for generic texture; otherwise, these fields shall be zero, FALSE, or blank as appropriate.

GTEXLIB	GTDB Texture Library Field
GTEXID	GTDB Texture ID Field
STAGE	Processing Stage Field
SGFLAG	Specific or Generic Texture Flag Field
TTYPE	Texture Type Field
TEXDES	Texture Description Field
HRES	Horizontal Resolution Field
VRES	Vertical Resolution Field
HSIZE	Horizontal Size Field
VSIZE	Vertical Size Field
MSTF	Modified Specific Texture Flag Field
NRF	Noise Removal Flag Field
ORF	Occlusion Removal Flag Field
HRF	Haze Removal Flag Field
SMF	Shadow Minimization Flag Field
IICEF	Inner Image Contrast Enhancement Flag Field
ITICEF	Image-to-Image Contrast Enhancement Flag Field
2GCF	Two-D Geometric Correction Flag Field
3GCF	Three-D Geometric Correction Flag Field
IQC	Image Quality Comment Field
IQR	Image Quality Rating Field
ICAPDT	Image Capture Date and Time Field
IFCRDT	Image File Creation Date and Time Field
LMDT	Last Maintenance Date and Time Field
PAST	Positional Accuracy Standards Field
GEOLOC	Geographic Location Name Field
GTSNAME	Generic Texture Set Name Field

5.14.2.2.2.1.2 **Source Data**. Valid source data shall be provided in all fields for all texture types, except for the SEID, SETYPE, and SENAME Fields. For these three fields, valid data shall be provided for stage 1 and stage 2 Areal and Model Texture only, and be zero otherwise. The PAST field shall contain valid data for all types except generic, for which it shall contain zero.

NUMDS	Number of Data Source Table Entries Field
	for each data source
SOID	Source ID Field
SOTYPE	Source Type Field
SONAME	Source Name Field
SOAP	Source Agency/Project Field
SODATE	Source Date Field
SEID	Sensor ID Field
SETYPE	Sensor Type Field
SENAME	Sensor Name Field
REDA	Reliability of Data Field
PAST	Positional Accuracy Standards Field
COLSYS	Collection System Field
CODATE	Compilation Date Field
SYNDF	Synthetic Data Flag Field
COMCRI	Compilation Criteria Field
ICAPDT	Image Capture Date and Time Field

5.14.2.2.2.1.3 **Environmental Conditions Data**. Valid data shall be provided for all Specific Areal and Model Texture. Zeroes and blanks shall be provided for all other texture types. Valid data shall also be provided for SMC/FDC texture in the SPENVC field.

SPENVC	Special Environmental Conditions Field
PERCC	Percent of Cloud Cover Field
PERSC	Percent of Shadow Cover Field

5.14.2.2.2.1.4 **Texture Footprint Data**. Valid data shall be provided as follows: PERTT and PERST - Stage 3 through Stage 5 Areal and Model Texture, SMC/FDC texture; NUMBOU, BOUNDID, SOID, NUMBP, BPID, and ICO - All Areal and Model Texture, SMC/FDC texture; LATLON - Areal and SMC/FDC texture; RELCO - Model Texture. Fields shall contain Zeroes, blanks, and FALSE values otherwise.

PERTT	Percent of Texture in Tile Field
PERST	Percent of Specific Texture Field
NUMBOU	Number of Boundaries Field
	for each boundary
BOUNDID	Boundary ID Field
SOID	Source ID Field
NUMBP	Number of Boundary Points Field
	for each boundary point
BPID	Boundary Point ID Field
LATLON	Latitude/Longitude Field (for Areal)
RELCO	Relative Coordinates Field (for Model)
ICO	Image Coordinates Field

5.14.2.2.2.1.5 **Neighbor Texture Association Data.** Valid data shall be provided in NOTNID, SOTNID, EATNID and WETNID Fields for SMC/FDC and Stage 3 through Stage 5 Areal Texture. Valid data shall be provided in ABTNID, BETNID, RITNID and LETNID Fields for Stage 3 through Stage 5 Model Texture. Fields shall be blank otherwise.

NOTNID	North Tile Neighbor ID Field
SOTNID	South Tile Neighbor ID Field
EATNID	East Tile Neighbor ID Field
WETNID	West Tile Neighbor ID Field
ABTNID	Above Tile Neighbor ID Field
BETNID	Below Tile Neighbor ID Field
RITNID	Right Tile Neighbor ID Field
LETNID	Left Tile Neighbor ID Field

5.14.2.2.2.1.6 **Model Association Data.** Valid data shall be provided for all specific model texture. Zero and blank fields shall be provided for all other texture types.

NUMMI	Number of Models in Image Field
	for each model
MODLIB	Model Library Type Field
MODNUM	Model Number Field
MODNAME	Model Name Field
MODVIEW	Model View Description Field

5.14.2.2.2.1.7 **Image Control Data.** Valid data shall be provided in these fields as follows: NUMCP, CPID, CPNAME, SOID and ICO(C) - Stage 1 and Stage 2 Areal and Model Texture; LATLON - Stage 1 and Stage 2 Areal Texture; RELCO, NUMMTP, MTPID, ICO(M), MODLIB, and MODNUM - Stage 1 and Stage 2 Model Texture; NUMGTP, GTPID and ICO(G) - Stage 1 and Stage 2 Areal Texture.

NUMCP	Number of Control Points Field
	for each control point
CPID	Control Point ID Field
CPNAME	Control Point Name Field
SOID	Source ID Field
LATLON	Latitude/Longitude Field (for Areal)
RELCO	Relative Coordinates Field (for Model)
ICO(C)	Image Coordinates Field (Control Point)
NUMGTP	Number of Generic Tie Points Field
	for each geographic tie point reference
GTPID	Geographic Tie Point ID Field
ICO(G)	Image Coordinates Field (Geographic Tie Point)
NUMMTP	Number of Model Tie Points Field
	for each model tie point reference
MTPID	Model Tie Point ID Field
ICO(M)	Image Coordinates Field (Model Tie Point)
MODLIB	Model Library Type Field
MODNUM	Model Number Field

5.14.2.2.2.1.8 **Sensor Image Descriptor Data.** Valid data shall be provided for all Stage 1 and Stage 2 Specific Areal and Specific Model Texture. Zero, blank, and FALSE values shall be provided for all other texture types.

NUMSEN	Number of Sensors Field
	for each sensor
SEID	Sensor ID Field
FILMQ	Film Quality Field
SUNAZ	Sun Azimuth Field
SUNEL	Sun Elevation Field
NUMSTM	Number of Stereo Mates Field
	for each stereo mate
GTEXID	GTDB Texture ID Field
SCANID	Scanner ID Field
SCRES	Scanner Resolution Field
SCFID	Scanner Filter ID Field
LLCOR	LL Corner X/Y Image Coordinates Field
ULCOR	UL Corner X/Y Image Coordinates Field
URCOR	UR Corner X/Y Image Coordinates Field
LRCOR	LR Corner X/Y Image Coordinates Field
CALFL	Calibrated Focal Length Field
CALPPO	Calibrated Principal Point Offset Field
CALPSO	Calibrated Point of Symmetry Offset Field
NUMFID	Number of Fiducial Coordinates Field
	for each fiducial coordinate
CALRIC	Calibrated Report Image Coordinates Field
MEAIC	Measured Image Coordinates Field
OMEGA	Omega Field
PHI	Phi Field
KAPPA	Kappa Field
RECTIF	Rectification Field
CAMPLL	Camera Position in Lat/Lon Field
CAMPH	Camera Position in Height Field
MSEOPK	Mean Square Error Omega/Phi/Kappa Field
MSELLH	Mean Square Error Lat/Lon/Height Field
HCAPTS	Horizontal Captured Texel Size Field
VCAPTS	Vertical Captured Texel Size Field

5.14.2.3 **Original Format Image File(s).** This file shall be included if Stage 1 Areal Texture is provided. Data shall be provided in its native format.

5.14.2.3.1 **Filename.** The ANSI filename for this file shall be "ssGnnnnnnOFIitttttt.y", where "ss" is the security code, "nnnnnn" is the GTDB identifier, "i" is "A" for the Areal Texture Library or "M" for the Model Texture Library, "ttttt" is the texture image number, and "y" is an alphabetic character assigned sequentially.

5.14.2.4 **NITF Image Data File.** The NITF Image Data File shall be included if non-Stage 1 Areal Texture is provided. Image data shall be formatted as stated in the NITF standard document.

5.14.2.4.1 **Filename.** The ANSI filename for this file shall be "ssGnnnnnnTEXitttttt.DAT", where "ss" is the security code, "nnnnnn" is the GTDB identifier, "i" is "A" for the Areal Texture Library or "M" for the Model Texture Library, and "ttttt" is the texture image number.

5.14.2.4.2 **File Format.** The format of this file shall be as follows:

```
if Image Mode is Band Sequential (BSQ)
   for each band
      for each block
         for each row from top to bottom
            for each column from left to right
               Texel Value

elseif Image Mode is Band Interleaved (BIL)
   for each block
      for each band
         for each row from top to bottom
            for each column from left to right
               Texel Value
```

5.15 **Model Texture (MT) Library.** The Model Texture (MT) Library is optional in a GTDB. The MT shall consist of files in the National Imagery Transmission Format (NITF) and files in their original native format. All NITF files shall follow a format as that described in the National Imagery Transmission Format (NITF) Version 1.1 document dated 1 March 1989 and the NITF Change Notice Number 2 dated 23 May 1990.

5.15.1 **MT File Structure.** The file order of the MT library shall be as follows:

```
NITF Header File
for each model texture
   NITF Image Sub-Header File
   if texture is in Stage 1 then
      Original Format Image File(s) [optional]
   else
      NITF Image Data File [optional]
   end if
```

5.15.2 **MT File Record Structures.** The record structure of each of these files shall be as described in the following subsections.

5.15.2.1 **NITF Header File.** This file shall be included if model texture is provided. The format for this file shall be identical to that in the AT Library.

5.15.2.2 **NITF Image Sub-Header File.** This file shall be included if model texture is provided. The format for this file shall be identical to that in the AT Library.

5.15.2.3 **Original Format Image File(s).** This file shall be included if Stage 1 Areal Texture is provided. This data shall be provided in its native format.

5.15.2.4 **NITF Image Data File.** This file shall be included if non-Stage 1 Areal Texture is provided. The format for this file in the MT Library is identical to that in the AT Library.

5.16 **Microdescriptors.** All microdescriptors included within a GTDB shall conform to the following format.

5.16.1 **GTDB Microdescriptor Records.** The code in parentheses is the attribute identifier which shall appear in the Microdescriptor Type field of a GTDB Microdescriptor Record.

5.16.1.1 **Homogeneous Area Microdescriptor Record (HA).** The attributes defined by this microdescriptor shall be as listed below.

(HA 01) Length of Individual Unit of Primary Field
(HA 02) Width of Individual Unit of Primary Field
(HA 03) SMC of Background Field

5.16.1.2 **Pattern Distribution Microdescriptor Record (PM).** The attributes defined by this microdescriptor shall be as listed below.

(PM 01) 1-D Separation Field
(PM 02) 2-D Length Portion Field
(PM 03) Angular Spatial (Radial) Field
(PM 04) Cartesian Orientation of Grid Field
(PM 05) Dimensionality Field
(PM 06) Distribution Field
(PM 07) Geometry Field
(PM 08) Radial Origin of Grid Field
(PM 09) Regularity of Grid Field
(PM 10) Width of Cell (Other) Field

5.16.1.3 **Drainage Microdescriptor Record (TTAD).** The attributes defined by this microdescriptor shall be as listed below.

(TTAD01) Bank Vegetation Field
(TTAD02) Bottom Material Field
(TTAD03) Gap Width Field
(TTAD04) Height, Left Bank, Field
(TTAD05) Height, Right Bank, Field
(TTAD06) Slope, Left Bank, Field
(TTAD07) Slope, Right Bank, Field
(TTAD08) SMC, Left Bank, Field
(TTAD09) SMC, Right Bank, Field
(TTAD10) Water Depth Field
(TTAD11) Water Velocity Field

5.16.1.4 **Transportation Microdescriptor Record (TTAT).** The attributes defined by this microdescriptor shall be as listed below.

(TTAT01) Bridge Bypass Condition Field
(TTAT02) Bridge Construction Material Field
(TTAT03) Bridge Horizontal Clearance Field
(TTAT04) Bridge Movement Field
(TTAT05) Bridge Overhead Clearance Field
(TTAT06) Bridge Type Field
(TTAT07) Depth of Overburden Field
(TTAT08) Highway Shoulder SMC Field
(TTAT09) Highway Shoulder Width Field
(TTAT10) Highway SMC Depth Field
(TTAT11) Highway Type Field
(TTAT12) Length Field
(TTAT13) Length of Spans Field
(TTAT14) Maximum Weight Field
(TTAT15) Minimum Horizontal Clearance Field
(TTAT16) Minimum Vertical Clearance Field
(TTAT17) Number of Spans Field
(TTAT18) Railroad Capacity Field
(TTAT19) Railroad Passing Tracks Field
(TTAT20) Railroad Type Field
(TTAT21) Reliability of Information Field
(TTAT22) Tracked Classification Field
(TTAT23) Transportation Qualifier Field
(TTAT24) Tunnel Height Field
(TTAT25) Underbridge Clearance Field
(TTAT26) Wheeled Classification Field
(TTAT27) Width Field

5.16.1.5 **Vegetation Microdescriptor Record (TTAV).** The attributes defined by this microdescriptor shall be as listed below.

(TTAV01) Cone Index Field
(TTAV02) Crown Diameter Field
(TTAV03) Depth of Bedrock Field
(TTAV04) Height of Lowest Branches Field
(TTAV05) Roughness Index, Foot Troops, Field
(TTAV06) Roughness Index, Large Tracked, Field
(TTAV07) Roughness Index, Large Wheeled, Field
(TTAV08) Roughness Index, Small Tracked, Field
(TTAV09) Roughness Index, Small Wheeled, Field
(TTAV10) SMC Depth Field
(TTAV11) Soil Qualifier Field
(TTAV12) Soil State Field
(TTAV13) Soil Type Field
(TTAV14) Stem Diameter Field
(TTAV15) Stem Spacing Field
(TTAV16) Summer Canopy Cover Field
(TTAV17) Vegetation Roughness Field
(TTAV18) Winter Canopy Cover Field

5.16.1.6 **Vertically Composite Microdescriptor Record (VC).** The attributes defined by this microdescriptor shall be as listed below.

(VC 01) Construction Type Code Field
(VC 02) Height of Elevated Portion Field
(VC 03) Length/Radius of Elevated Portion Field
(VC 04) Placement Code Field
(VC 05) Shape Code of Elevated Portion Field
(VC 06) SMC for Elevated Portion of Feature Field
(VC 07) Width of Elevated Portion Field

5.16.2 **Temporal Effects Microdescriptor Records.** Temporal effects microdescriptors shall conform to the following format. The Microdescriptor Type Field shall contain the value specified in parentheses in the subparagraphs.

Microdescriptor Type Field
Temporal Condition Field
Number of Attributes Affected Field

5.16.2.1 **Weather Effects Microdescriptor Record (TEW).** The temporal conditions defined by this microdescriptor shall be as listed below.

(TEW 01) Alternate Temperature Threshold Field
(TEW 02) Alternate Precipitation Condition Field
(TEW 03) Alternate Cloud Cover Condition Field

5.16.2.2 **Seasonal Effects Microdescriptor Record (TES).** The temporal conditions defined by this microdescriptor shall be as listed below.

(TES 01) Alternate Season Field

5.16.2.3 **Time of Day Microdescriptor Record (TET).** The temporal conditions defined by this microdescriptor shall be as listed below.

(TET 01) Alternate Time of Day Threshold Field

5.16.2.4 **Ground Conditions Microdescriptor Record (TEG).** The temporal conditions defined by this microdescriptor shall be as listed below.

(TEG 01) Alternate Ground Condition Field

5.16.2.5 **Alternate Attributes Record (TEAA).** One or more TEAA records shall follow each occurrence of a TEW, TES, TET, or TEG record. The number of TEAA records shall correspond to the Number of Attributes Affected field within the parent temporal effects record. The fields within this microdescriptor record shall be as listed below.

Microdescriptor Type Field (always 'TEAA ')
Affected Attribute Code Field (e.g., 'TTAD10')
Alternate Attribute Value Field

5.17 **Feature Codes and Attributes.** Features included in a GTDB shall conform to modified version of the DMA FACS standard, as defined in Appendix B of this standard.

6 NOTES

(This section contains information of a general or explanatory nature that may be helpful, but is not mandatory.)

6.1 **Intended Use.** This standard is intended to be used for the interpretation of the format of the Generic Transformed Data Base.

6.2 **Acquisition Requirements.** Acquisition documents must specify the following:

a. Title, number, and date of the specification.

b. Issue of DoDISS to be cited in the solicitation, and if required, the specific issue of individual documents referenced (see 2.2).

6.3 **Subject term (key word) listing.**

Data Base

6.4 **Referenced documents.** The following documents were used as references, in preparation of this DBDD.

ANSI/MIL-STD-1815A, Ada Programming Language.

Defense Mapping Agency Digital Landmass System Product Specification, First Edition, July 1977.

Defense Mapping Agency Digital Landmass System Product Specification, Second Edition, April 1983.

Defense Mapping Agency Product Specifications for a Prototype Data Base to Support High Resolution Sensor Simulation, First Edition, December 1979.

Defense Mapping Agency Prototype High Resolution Data Base Product Specification, First Edition, August 1980.

Defense Mapping Agency Level X Product Specification, First Edition, June 1983.

Defense Mapping Agency Feature File Product Specification, First Edition, August 1984.

Defense Mapping Agency Standard Linear Format Product Specification, First Edition.

Defense Mapping Agency (DMA) High Resolution Data (Level X) Specification for B-1B Simulator.

DMA Standard Supporting Mark 90, Section 100, Glossary of Feature/Attribute Definitions, Second Edition, June 1988, revised December 1988.

CUSTODIANS:
Army - PT
Navy - TD
Air Force - 11

PREPARING ACTIVITY:
Air Force - 11

(Project Nr. 69GP-0101)

APPENDIX A

This page intentionally left blank.

DATA DICTIONARY

10 **SCOPE.** This appendix is an alphabetical data dictionary defining every field which occurs within a GTDB.

20 APPLICABLE DOCUMENTS

This section does not apply.

30 DEFINITIONS AND ACRONYMS

This section does not apply.

40 GENERAL REQUIREMENTS

40.1 This Appendix shall be a mandatory part of the standard. The information contained herein is intended for compliance.

50 DETAILED REQUIREMENTS

50.1 . Each dictionary entry includes the following items of information:

- fieldname, as listed in the GTDB record definitions in section 3.3 of the GTDB DBDD;
- data type, indicating whether the data value is represented logically as an integer, a real number, a Boolean, an Ada enumerated type, a text string, or a composite record of subsidiary fields;
- length of the field, in ASCII characters;
- range of valid values;
- a brief narrative description of the field.

50.2 All fields on a GTDB tape are encoded in ASCII character format, except for those in the NITF Image Data Files and the Original Format Image Files. The logical interpretation of the ASCII characters varies with the information contained under "Type" and "Length". Table A-I is a key to the abbreviations contained under "Type" and to the format of the ASCII strings contained in the various types of fields for non-NITF files.

TABLE I

Type	Length	Meaning	ASCII Format
B	1	Boolean, True/False	'T' or 'F'
E	any	Ada Enumerated Type	alphanumeric text
H	N	Hexidecimal number of N characters	hhhhhhhhhh etc. for N characters
I	2	Integer, 1 digit	sn, where 's' is a minus sign (or blank) and 'n' is a digit
I	4	Integer, 3 digits	snnn, where 's' is the sign and 'n' is a digit
I	6	Integer, 5 digits	snnnnn
I	7	Integer, 6 digits	snnnnnn, where 's' is a minus sign (or blank) and 'n' is a digit
I	11	Integer, 10 digits	snnnnnnnnnn
R6	12	Real, 6 significant digits	sn.nnnnnEsee, where 'ee' is the exponent
R10	16	Real, 10 significant digits	sn.nnnnnnnnnEsee
I2D	L	Integer pair	two integers separated by ASCII null characters; each integer has a length of the maximum number of digits plus 1 for the sign; thus, the total length L = L1 + L2 + 1 where L1 and L2 are the maximum lengths of the first and second integers, respectively.

TABLE I

Type	Length	Meaning	ASCII Format
I3D	L	Integer triplet	three integers separated by ASCII null characters; each integer has a length of the maximum number of digits plus 1 for the sign; thus, the total length L = L1 + L2 + L3 + 2 where L1, L2, and L3 are the maximum lengths of the first, second, and third integers, respectively.
I4D	L	Integer quad	four integers separated by ASCII null characters; each integer has a length of the maximum number of digits plus 1 for the sign; thus, the total length L = L1 + L2 + L3 + L4 + 3 where L1, L2, L3, and L4 are the maximum lengths of the first, second, third, and fourth integers, respectively.
R2D6	25	Real pair, 6 significant digits	two real numbers of the R6 type separated by ASCII null characters
R3D6	38	Real triplet, 6 significant digits	three real numbers of the R6 type separated by ASCII null characters
R4D6	51	Real quad, 6 significant digits	four real numbers of the R6 type separated by ASCII null characters
R2D10	33	Real pair, 10 significant digits	two real numbers of the R10 type separated by ASCII null characters
R3D10	50	Real triplet, 10 significant digits	three real numbers of the R10 type separated by ASCII null characters
S	any	Text String	alphanumeric text

50.2.1 Note that the size of each ASCII field is fixed to contain enough character positions to hold the largest valid value for that field, including a sign for all numerics. Data values within all fields will be right-justified, with blank padding to the left for text.

50.2.2 Within a record in a non-NITF file, all fields are separated from each other by an ASCII null character. This makes it possible to process specific fields without knowledge of the format and length of intervening fields. Users are encouraged to make use of this feature, as it isolates the user from changes to the format of fields not relevant to the user's application.

50.2.3 NITF files follow the NITF format rules, where every field with a value is preceded by a field with a field label for that value. Field separators for these files are defined by NITF as a carriage return character followed by a line feed character. Three main differences between the types of the NITF files and the types of the non-NITF files are that (1) NITF Boolean Types are spelled out ("TRUE", "FALSE") while the non-NITF files simply use "T" and "F"; (2) the NITF Integer Types are of a length equal to the maximum length number, including a sign character only if the number can be negative, while the non-NITF Integer types have a fixed set of lengths that always include a character for the sign, regardless of whether that specific field can be negative; and (3) for composite fields (fields with multiple data items), the intra-field separator is the ASCII blank character, not the ASCII null character used in non-NITF files in a GTDB. The following tables are a key to the abbreviations contained under "Type" and to the format of the ASCII strings contained in the various types of fields for NITF files.

TABLE II

Type	Length	Meaning	ASCII Format
B	5	Boolean, True/False	'TRUE' or 'FALSE'
E	any	Ada Enumerated Type	alphanumeric text
I	L	Integer	sN, where 's' is optional and if it is there, it is a minus sign, and N is a series of digits where the number of digits is the maximum number required to specify the largest possible number in the valid range. If not all the digits are required, the integer is padded with zeroes to the left.
R6	12	Real, 6 significant digits	sn.nnnnnEsee, where 'ee' is the exponent
R10	16	Real, 10 significant digits	sn.nnnnnnnnnEsee
I2D	L	Integer pair	two integers separated by ASCII blank characters; each integer has a length of the maximum number of digits; thus, the total length L = L1 + L2 + 1 where L1 and L2 are the maximum lengths of the first and second integers, respectively.

TABLE II

Type	Length	Meaning	ASCII Format
I3D	L	Integer triplet	three integers separated by ASCII blank characters; each integer has a length of the maximum number of digits; thus, the total length L = L1 + L2 + L3 + 2 where L1, L2, and L3 are the maximum lengths of the first, second, and third integers, respectively.
R2D6	25	Real pair, 6 significant digits	two real numbers of the R6 type separated by ASCII blank characters
R3D6	38	Real triplet, 6 significant digits	three real numbers of the R6 type separated by ASCII blank characters
R4D6	51	Real quad, 6 significant digits	four real numbers of the R6 type separated by ASCII blank characters
R2D10	33	Real pair, 10 significant digits	two real numbers of the R10 type separated by ASCII blank characters
R3D10	50	Real triplet, 10 significant digits	three real numbers of the R10 type separated by ASCII blank characters
S	any	Text String	alphanumeric text
BIN	--	Binary for texel values	--

TABLE A-III

Fieldname	Type	Length	Range	Description
Above Tile Neighbor ID (NITF)	I	10	0..2147483647	The identifier of the neighboring model specific image above the current image.
Absorptivity	R6	12	0.0..1.0	Ratio of radiant (thermal) energy absorbed by a feature to the energy incident upon it. Synthetic default values are generated by P2851 software as a random function of Surface Material Category.
Area Block Boundary	I2D, I2D	43	-324000000.. 324000000, -648000000.. 648000000; -324000000.. 324000000, -648000000.. 648000000	A pair of latitude/longitude coordinates defining the southwest and northeast corners of an area block. See definitions for Lat/Long SW Corner and Lat/Long NE Corner. The coordinates are separated by ASCII null characters.
Area Block Category	E	18	AREAL, LINEAR, POINT, POINT_LIGHT, POINT_LIGHT_STRING, MODEL_REF, TERRAIN_GRID, TERRAIN_POLY, VERTEX	Type of data contained within an area block pseudo-file.

TABLE A-III

Fieldname	Type	Length	Range	Description
Area Block ID	I, I, E	34	1..16; 1..2147483647; AREAL, LINEAR, POINT, POINT_LIGHT, POINT_LIGHT_STRING, MODEL_REF, TERRAIN_GRID, TERRAIN_POLY, VERTEX	A unique identifier for an area block, consisting of a SLOD ID, an Area Block Number, and an Area Block Category.
Area Block Number	I	11	1..2147483647	Ten-digit sequence number used to uniquely identify an area block within a GTDB.
Area Block Size	I2D	21	-324000000.. 324000000, -648000000.. 648000000	The dimensions of an area block, expressed in thousandths of seconds of relative latitude and longitude. See definitions for Relative Latitude and Relative Longitude. The two fields are separated by an ASCII null character.
Areal Feature Area Block Flag	B	1	T, F	Boolean flag indicating whether an Areal Feature Area Block (AFAB) pseudo-file exists within the given area block.
Articulated Part Flag	B	1	T, F	Indicates whether the referenced model is an articulated part.
Attachment (NITF)	I	3	0..998	Display level to which a new object is to be attached for editing purposes.

TABLE A-III

Fieldname	Type	Length	Range	Description
Attribute Value	--	--	--	The value assigned to a FACS code. This value will differ based on the type of FACS code. The following list shows the values that can be assigned: one byte integer, two byte integer, four byte integer, real (six significant digits), long real (ten significant digits), integer pair (2 - four byte integers), integer triplet (3 - 4 byte integers), real pair (2 real numbers with six significant digits), real triple (3 real numbers with six significant digits), long real pair (2 real numbers with ten significant digits), long real triple (3 real numbers with ten significant digits), string, enumerated, boolean (flag), or null value; the length values are:

Class	Length
INT1	4
INT2	6
INT4	11
INT2D	23
INT3D	35
REAL6	12
REAL10	16
REAL2D6	25
REAL2D10	33
REAL3D6	38
REAL3D10	50
FLAG	5
STR	up to 80
ENUM	up to 80
NONE	0

TABLE A-III

Fieldname	Type	Length	Range	Description
Base Polygon ID	I	11	0..2147483647	Polygon ID of a polygon within a model which also serves to define the footprint of that model on the ground.
Below Tile Neighbor ID (NITF)	I	10	0..2147483647	The identifier of the neighboring model specific image below the current image.
Bits Per Texel Per Band	I	4	1..64	The number of data bits for each texel for each band in the image before compression. For multi-band images treated as a single image, the number of bits per texel is identical for each band. Standard values include 1, 8, and 16 for intensity, color, or other multispectral textures; 56 for SMC/FDC textures, and 16 or 24 for terrain. Same as Number of Bits Per Pixel Per Band for NITF data.
Blending Type	I	6	0..32767	Indicates type of color blending to apply to a feature. (TBD)
Boundary ID (non-NITF)	I	6	0..32767	A unique identifier for a terrain or texture footprint.
Boundary ID (NITF)	I	5	0..32767	A unique identifier for a terrain or texture footprint.

TABLE A-III

Fieldname	Type	Length	Range	Description
Boundary Point	I2D	21	-324000000.. 324000000, -648000000.. 648000000	A latitude/longitude coordinate defining a point on the boundary of the gaming area, or of an island within the gaming area. See definitions for Latitude and Longitude. The two values are separated by an ASCII null.
Boundary Point ID (NITF)	I	5	0..32767	A unique identifier for a point that is located on a terrain or texture footprint.
Boundary Point Count	I	11	5..2147483647	The number of Boundary Point records used to define the boundaries of the gaming area.
Calibrated Focal Length (NITF)	R10	16	10.0..10000.0	An adjusted value of the focal length computed to equalize the positive and negative values of lens distortion over the entire focal plane (expressed in millimeters).
Calibrated Point of Symmetry Offset (NITF)	R10	16	-1.0..1.0	The adjusted position which gives the best symmetry of radial lens distortion. The indicated principal point is formed by the intersection of lines connecting the fiducial marks. The calibrated point of symmetry offset is the difference between these two points (expressed in meters).

TABLE A-III

Fieldname	Type	Length	Range	Description
Calibrated Principal Point Offset (NITF)	R10	16	-1.0..1.0	The foot of the perpendicular from Offset the interior perspective center to the focal plane. The indicated principal point is formed by the intersection of lines connecting the fiducial marks. The calibrated principal point offset is the difference between these points (expressed in meters).
Calibrated Report Image Coordinates (NITF)	R2D10	33	-1.0..1.0, -1.0..1.0	The position of the fiducial marks as determined by a laboratory calibration of the camera (expressed in meters).
Camera Position in Height (NITF)	R10	16	0.0.. 9.999999999E+99	Height of the camera above mean sea level expressed in meters.
Camera Position in Lat/Lon (NITF)	S	24	HDDMMSSSSSb HDDDMMSSSSS	Geographic location of the sensor used to capture the data expressed in thousandths of arc seconds in absolute coordinates; H = hemisphere, DD or DDD = degrees, MM = minutes, SSSSS = thousandths of seconds, and b = blank character (" ")
Centroid	R2D6	25	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	A coordinate defining the center of a circumscribing circle or sphere for a feature, model, or model polygon; or the center of curvature for a point light string. The internal format of this field varies with the user-specified coordinate system for the GTDB.
	R3D6	38	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25, -1.93428E+25, 1.93428E+25	

TABLE A-III

Fieldname	Type	Length	Range	Description
Checksum	I	6	0..32767	The logical sum of all characters contained within a file or pseudo-file in a GTDB, used to verify that data have been correctly read from the input medium.
Chroma	I	6	0..32767	The chroma component of a color defined by the Hue-Chroma-Value model. P2851 normalizes Chroma to a range of 0 (grey) to 32767 (color) instead of the familiar 0-100 percent saturation range. Synthetic default values are generated by P2851 software as a random function of Surface Material Category.
Cluster ID (Model Cluster ID)	H	32	0..F (32 times)	In a model, a number identifying a set of polygons which have been grouped by definition of separation planes. It is a 32-digit hexadecimal number, supporting 128 separation planes in a model.
Collection System (NITF)	S	10	--	ID of the system used to collect data.
Color	I4D	25	0..32676, 0..32767, 0..32767, 0..127	Numerical representation of color in terms of hue, chroma, and value, along with an index into a P2851 Color Calibration Table. See definitions for Hue, Chroma, Color Value, and Color Calibration Entry ID. The four values will be separated by ASCII nulls.

TABLE A-III

Fieldname	Type	Length	Range	Description
Color Calibration Entry ID	I	4	1..127	An index to an entry in the P2851 Color Calibration Table, based on relative position of the entry.
Color Characteristics				Same as Color.
Color ID	I	4	0..255	Code designating an approximate color, based on a limited standard palette. A value of '0' indicates unknown.
Color Name	E	6	RED, ORANGE, YELLOW, GREEN, BLUE, INDIGO, VIOLET	Common color name approximating the color represented numerically in the Color field.
Color Table Index	I	11	0..2147483647	The name of the color table file included with SIF/HDI models or SIF/HDI culture data.
Color Texture Existence Flag	B	1	T, F	The flag indicating the existence of color texture within an area block or an island LOD.
Color Texture Resolution	R6	12	0.0..1.93428e+25	Resolution of color texture in thousandths of arc-seconds/texel for geodetic textures and meters/texel for all other textures.

TABLE A-III

Fieldname	Type	Length	Range	Description
Color Value	I	6	0..32767	The value component of a color defined by the Hue-Chroma-Value model. P2851 normalizes Value to a range of 0 (black) to 32767 (white) instead of the familiar 0-100 percent lightness range. Synthetic default values are generated by P2851 software as a random function of Surface Material Category.
Column Count	I	7	1..108000	The number of Terrain Column records used to store elevation values within a TGAB.
Compilation Criteria (NITF)	S	160	--	Description of data capture criteria.
Compilation Date (NITF, non-NITF)	S	6	YYMMDD	The date on which a GTDB was generated Any succeeding subarea updates will leave this field unchanged but will be reflected in the Last Update Date.
Component ID	I	7	0..1000	Unique ID number assigned to components of a model.

TABLE A-III

Fieldname	Type	Length	Range	Description
Compression Rate Code (NITF)	S	4	"C0 ", "C1 ", "C2 ", "NC "	A code indicating the compression rate for the image. If Image Compression = C0, the code is user defined. If Image Compression = C1, the codes are as follows: 1D = 1 Dimensional Coding; 2DS = 2 Dimensional Coding Standard Vertical Resolution, K=2; 2DH = 2 Dimensional Coding High Vertical Resolution, K=4. If Image Compression = C2, the Compression Rate Code is given in the form n.nn, representing the average number of bits per pixel over the image after compression. Valid codes are 0.75, 1.40, 2.30, and 4.50. Omitted if Image Compression is NC.
Control Point ID (NITF)	I	5	0..32767	Numeric identifier of a specific control point.
Control Point Name (NITF)	S	40	--	Field used to associate a textual name with a control point.
Coordinate	R2D6	25	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	A representation of a point in 2D space used to define model vertices.
	R3D6	38	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	A representation cf a point in 3D space used to define model vertices.

TABLE A-III

Fieldname	Type	Length	Range	Description
Coordinate	R3D10	50	-9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99	A representation of a point in longitude, latitude, and elevation. Longitude and latitude are in thousandths of arc seconds when the coordinate system is GEODETIC_FLOAT; otherwise, they are in meters. Elevation is always in meters. Used to define terrain posts in gridded terrain and vertices in polygonized terrain and culture.
Coordinate System	E	14	GEODETIC_FLOAT, GEOCENTRIC, MERCATOR, TRANS_MERC, LAMBERT, POLAR, LOCAL	User-defined CDBTP parameter specifying which of the allowable coordinate systems should be used to represent terrain and culture vertices within the GTDB.
Correlation Priority	I	4	0..127	A number indicating the relative importance of a feature or vertex for maintaining correlation among GTDBs generated to support different simulators. Higher numbers indicate greater priority.
Correlation Priority (Vertex List Pointer Record)	I	2	0..9	A number indicating the relative importance of a model vertex for maintaining correlation among GTDBs generated to support different simulators. Higher numbers indicate greater priority.

TABLE A-III

Fieldname	Type	Length	Range	Description
Culture Resolution	R10	16	-9.999999999E+99.. 9.999999999E+99	User-defined CDBTP parameter giving the maximum perpendicular error to be introduced by the culture thinning process. A value of 0.0 indicates that thinning should not be performed.
Cycle Rate Off Time	I	11	0..2147483647	The period of time which a light remains in the off state, expressed in thousandths of seconds.
Cycle Rate On Time	I	11	0..2147483647	The period of time which a light remains in the on state, expressed in thousandths of seconds.
Data Base Sentinel (NITF)	S	4	"GTDB"	P2851 Data Base type identifier used with NITF data
Data Location	E	8	ON_LINE, OFF_LINE	Indicator used by P2851 software for configuration management. The field may be ignored by users.

TABLE A-III

Fieldname	Type	Length	Range	Description
Datum ID	E	26	WGS_84, WGS_72, NONE, BOGOTA_OBSERVATORY, CHATHAM_1971, CHUA_ASTRO, CORREGO_ALEGRE, EUROPEAN_1950, GEODETIC_DATUM_1949, HJORSEY_1955, HONG_KONG_1963, PROVISIONAL_SOUTH_AMERICAN, QORNOQ, ROME_1940, SOUTH_AMERICAN_1969, BERMUDA_1957, CAPE_CANAVERAL, L_C_5_ASTRO, LUZON, NORTH_AMERICAN_1927, OLD_HAWAIIAN, ARINDAN, ARC_1950, ARC_1960, LIBERIA_1964, MAHE_1971, MERCHICH, NAHRWAN, OMAN, SOUTH_EAST_ISLAND, VITI_LEVU_1916	A user-defined CDBTP parameter indicating the geodetic datum to be used for coordinate transformations.

TABLE A-III

Fieldname	Type	Length	Range	Description
Datum Shift	R3D10	50	-9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99	A user-defined CDBTP parameter indicating the datum shift values to be applied during coordinate transformations. (Only applies if Datum ID is NONE.)
Decompose Culture Flag	B	1	T, F	User-defined CDBTP parameter indicating whether concave areal features (i.e., those having one or more exterior angles less than 180 degrees) should be decomposed into convex polygons none of whose exterior angles are less than 180 degrees.
Diffuse Reflectance	R6	12	0.0..1.0	Radar backscatter coefficient, expressed as a ratio. Synthetic default values are generated by P2851 software as a random function of Surface Material Category.
Directionality	R10	16	0.0..361.0	Angle from north from which a point light feature is visible. A value of '361.0' indicates the light is omni-directional.
Directivity (Infrared)	E	4	UNI, BI, OMNI	Indicator of shape of the planar response curve of a feature to infrared sensors. Synthetic default values are generated by P2851 software as a function of Feature Descriptor Code.

TABLE A-III

Fieldname	Type	Length	Range	Description
Directivity (Radar)	E	4	UNI, BI, OMNI	Indicator of shape of the planar response curve of a feature to radar sensors. Synthetic default values are generated by P2851 software as a function of Feature Descriptor Code.
Directivity (Visual)	E	4	UNI, BI, OMNI	Indicator of shape of the planar response curve of a feature to visual sensors. Synthetic default values are generated by P2851 software as a function of Feature Descriptor Code.
Directivity				Same as Directivity (Visual).
Display Level (NITF)	I	3	0..999	Unique graphic display level of an image relative to other message components in a composite. A higher number means that the item is to be displayed in front of other items with lower display level values.
East Tile Neighbor ID (NITF)	I	10	0..2147483647	The identifier of the neighboring areal specific image to the east.
Eccentricity	R10	16	-9.999999999E+99.. 9.999999999E+99	A user-defined CDBTP parameter applied during coordinate transformations. (Only applies if Datum ID is NONE.)
Elevation Reference	E	9	GEOID, ELLIPSOID	A user-defined CDBTP parameter indicating the reference level relative to which all elevation values are to be expressed.

TABLE A-III

Fieldname	Type	Length	Range	Description
Emissivity	R6	12	0.0..1.0	Ratio of the rate of infrared radiation from a feature or model as a consequence of its temperature only, to the corresponding rate of emission from a blackbody at the same temperature.
Encryption (NITF)	I	1	0..1	Flag that indicates whether an image is encrypted. 0 = no encryption, 1 = encrypted.
End Vertex ID	I	11	0..2147483647	The highest number assigned to a vertex within a Vertex Area Block pseudo-file.
Ending FDC Code	S	5	--	The Feature Descriptor Code used to define the end of a range of FDCs within various parameters of the CDBTP.
Exitance	R6	12	0.0..1.93428E+25	Rate of flow of infrared radiation from a feature or model polygon per unit of surface area.
Expand Lineals Flag	B	1	T, F	User-defined CDBTP parameter indicating whether linear features should be replaced by areal representations.

TABLE A-III

Fieldname	Type	Length	Range	Description
Extended Header Data (NITF)	--	--	--	Reserved for future use.
Extended Header Data Length (NITF)	I	5	0..99999	The length in bytes of the Extended Header Data Record.
Extended Sub-Header Data (NITF)	--	--	--	Reserved for future use.
Extended Sub-Header Data Length (NITF)	I	5	0..99999	The length in bytes of the Extended Sub-Header Data Record.
Face-Based Mapping Flag	B	1	T, F	Parameter indicating the existence of face-based mapping parameters for texture.
FACS Attribute Code	S	6	See Section 6.1	A code uniquely identifying an attribute type, based on codes defined in the DMA Feature Attribute Coding Standard.

TABLE A-III

Fieldname	Type	Length	Range	Description
FACS Class	E	8	INT1, INT2, INT4, INT2D, INT3D, REAL6, REAL10, REAL2D6, REAL2D10, REAL3D6, REAL3D10, STR, ENUM, FLAG, NONE	Specifies the data type based on which a FACS code is defined. INT1 represents integer type of one byte. INT2 represents integer type of two bytes. INT4 represents integer type of four bytes. INT2D represents a 2-D integer pair where the elements are of INT4 type. INT3D represents a 3-D integer triplet where the elements are of INT4 type. REAL6 represents real type with six significant digits. REAL10 represents real type with ten significant digits. REAL2D6 represents a 2-D real pair where the elements are of REAL6 type. REAL2D10 represents a 2-D real pair where the elements are of REAL10 type. REAL3D6 represents a 3-D real triplet where the elements are of REAL6 type. REAL3D10 represents a 3-D real triplet where the elements are of REAL10 type. STR represents string type. ENUM represents enumerated type. FLAG represents boolean type. NONE indicates that the user-defined FACS has no associated value.
Feature Descriptor Code	S	5	See section 6.1	Alphanumeric code assigned to classify a feature within a set of hierarchical categories, based on the DMA FACS as extended by P2851.
Feature Fragment Flag	B	1	T, F	Boolean flag indicating whether a feature has been clipped along an area block or terrain polygon boundary.

TABLE A-III

Fieldname	Type	Length	Range	Description
Feature Number	I	11	1..2147483647	Numeric ID unique to every feature within an area block.
Feature Onset	B	1	T, F	Indicator for changing radar backscatter coefficients. Synthetic default values are generated by P2851 software as a function of Feature Descriptor Code.
Feature Type	E	18	AREAL, LINEAR, POINT, POINT_LIGHT, POINT_LIGHT_STRING	Classifies a feature as areal, linear, point, point light, or point light string.
FID Code	S	6	NONE, FID101, FID102, FID103, FID104, FID105, FID106, FID110, FID111, FID112, FID113, FID114, FID115, FID116, FID120, FID121, FID122, FID130, FID135, FID136, FID137, FID138, FID145, FID146, FID150, FID151, FID152, FID153, FID154, FID155, FID156, FID160, FID161, FID162,	DMA FID for feature, if input from DMA DFAD source. For details of the meanings of these codes, refer to "DMA Product Specifications for DLMS Data Base", 2nd Edition, 1983.

TABLE A-III

Fieldname	Type	Length	Range	Description
FID Code	S	6	FID163, FID164, FID165, FID166, FID170, FID171, FID172, FID173, FID174, FID180, FID181, FID182, FID183, FID184, FID185, FID186, FID187, FID188, FID189, FID190, FID201, FID202, FID203, FID204, FID205, FID206, FID207, FID208, FID209, FID220, FID221, FID222, FID223, FID224, FID230, FID231, FID232, FID233, FID234, FID235, FID236, FID237, FID238, FID239, FID240, FID244, FID245, FID250, FID251, FID252, FID253, FID254, FID255, FID260, FID261, FID262, FID263, FID264, FID265, FID267, FID270, FID271, FID272, FID273, FID274, FID275, FID276, FID277, FID280, FID281, FID282, FID283, FID290, FID301, FID302, FID303,	DMA FID for feature, if input from DMA DFAD source. For details of the meanings of these codes, refer to "DMA Product Specifications for DLMS Data Base", 2nd Edition, 1983.

TABLE A-III

Fieldname	Type	Length	Range	Description
FID Code	S	6	FID304, FID305,	DMA FID for feature, if input from DMA DFAD source. For details of the meanings of these codes, refer to "DMA Product Specifications for DLMS Data Base", 2nd Edition, 1983.
			FID320, FID321,	
			FID322, FID323,	
			FID324, FID325,	
			FID330, FID331,	
			FID332, FID334,	
			FID340, FID341,	
			FID343, FID344,	
			FID350, FID352,	
			FID401, FID402,	
			FID403, FID420,	
			FID421, FID430,	
			FID433, FID434,	
			FID435, FID436,	
			FID450, FID451,	
			FID501, FID511,	
			FID512, FID520,	
			FID521, FID530,	
			FID531, FID532,	
			FID535, FID536,	
			FID540, FID541,	
			FID542, FID543,	
			FID544, FID560,	
			FID561, FID601,	
			FID602, FID603,	
			FID604, FID605,	
			FID606, FID610,	
			FID620, FID621,	
			FID622, FID630,	
			FID631, FID632,	
			FID640, FID641,	
			FID650, FID680,	
			FID681, FID682,	
			FID683, FID684,	
			FID701, FID702,	
			FID703, FID704,	
			FID705, FID706,	
			FID707, FID710,	

THIS

PAGE

IS

MISSING

IN

ORIGINAL

DOCUMENT

A-29

TABLE A-III

Fieldname	Type	Length	Range	Description
FID Code •	S	6	FID926, FID927, FID928, FID929, FID930, FID931, FID932, FID933, FID934, FID940, FID941, FID942, FID943, FID944, FID945, FID946, FID947, FID948, FID949, FID950, FID951, FID952, FID953, FID954, FID955, FID956, FID960, FID961, FID962, FID963, FID964, FID965, FID966, FID967, FID980	DMA FID for feature, if input from DMA DFAD source. For details of the meanings of these codes, refer to "DMA Product Specifications for DLMS Data Base", 2nd Edition, 1983.
File Name	S	80	--	The name of a data file which is located on the GTDB tape.
Film Quality (NITF)	S	20	--	Quality of film used to capture the image.
First Standard Parallel	R10	16	-90.0..90.0	A user-defined CDBTP parameter indicating a first standard parallel to be applied during projection transformations.
Fragment Culture Flag	B	1	T, F	User-defined CDBTP parameter indicating whether areal and linear features should be clipped along the boundaries of underlying terrain polygons.

TABLE A-III

Fieldname	Type	Length	Range	Description
Fragment Point Light Strings Flag	B	1	T, F	User-defined CDBTP parameter indicating whether point light string features should be fragmented into individual point light features.
Generic Model Flag	B	1	T, F	Indicates whether the model is generic.
Generic Texture Set Name (NITF)	S	20	--	Textual identifier identifying a set of generic textures that represent the same entity where each member of the set has a different size and/or resolution.
Geographic Location Name (NITF)	S	40	--	A textual name associated with an areal specific image or SMC/FDC image.
Geographic Tie Point ID (NITF)	I	10	0..2147483647	A unique identifier of a geographic tie point.
Global-Based Mapping Flag	B	1	T, F	Parameter indicating the existence of global-based mapping parameters for texture.
Global Reference Point	R2D10	33	-9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99	A point on culture/terrain which corresponds to the origin of the texture being mapped

TABLE A-III

Fieldname	Type	Length	Range	Description
Goodness-of-Fit	R10	16	-9.999999999E+99.. 9.999999999E+99	User-defined CDBTP parameter specifying the maximum allowable difference between any source elevation value and the corresponding elevation as represented by the surface of a GTDB terrain polygon. This parameter may be specified on a per area block basis.
Grayscale Texture Existence Flag	B	1	T, F	The flag indicating the existence of grayscale (or intensity) texture within an area block or an island LOD.
Grayscale Texture Resolution	R6	12	0.0..1.93428e+25	Approximate resolution of grayscale (or intensity) texture in units of Meters/texel.
GTDB Directory	S	80	--	Logical name of the P2851 VAX/VMS directory used in GTDB creation. This field is used by P2851 for configuration management and may be ignored by the user.
GTDB Texture ID (NITF)	I	10	0..2147483647	ID number assigned to a texture; unique within a GTDB texture library.
GTDB Texture Library (NITF)	E	5	AREAL, MODEL	ID of one of two GTDB texture libraries.
GTDB Texture Library Type	E	5	AREAL, MODEL	Same as GTDB Texture Library
GTDB User Defined Header Data Length (NITF)	I	5	00000..99999	Length in bytes of data defined specifically for the GTDB that is not part of the standard NITF data.

TABLE A-III

Fieldname	Type	Length	Range	Description
GTDB User Defined Image Data Length (NITF)	I	5	C0000..99999	Length in bytes of data defined specifically for the P2851 GTDB but not defined in the standard NITF Image Sub-Header.
GTDB Version Number	I	6	1..99999	The version number of a given GTDB, incremented each time it is updated or extended. A GTDB over the same gaming area but created from scratch would be considered a new GTDB rather than a new version of the previous GTDB.
Haze Removal Flag (NITF)	B	5	TRUE, FALSE	Flag indicating whether haze has been removed from an image.
High Value Field	R	6	0.0..100.0	Percentage.
Horizontal Captured Texel Size (NITF)	R10	16	0.0.. 9.999999999E+99	Approximate ground distance for a texel (expressed in meters) in the horizontal x-direction.
Horizontal Resolution (NITF, non-NITF)	R6	12	0.0..1.93428e+25	Horizontal length of a texel in meters, (e.g., 1.0 M/texel)
Horizontal Size (NITF)	R6	12	0.0..1.93428e+25	The horizontal size of the entire image, e.g., 1000.0 Meters
Horizontal Tile Block Size	I	11	0..2147483647	The number of texels in the horizontal direction (row) in a texture block. A texture is divided into a whole number of blocks, each consisting of the same dimensions.

TABLE A-III

Fieldname	Type	Length	Range	Description
Hue	I	6	0..32767	The hue component of a color defined by the Hue-Chroma-Value model. P2851 normalizes Hue to a range of 0 (blue) to 10922 (red) to 21845 (green) and back to 32767 (blue), instead of the familiar 360-degree color wheel, which turns from 0 (blue) to 120 (red) to 240 (green) back to 360 (blue). Synthetic default values are generated by P2851 software as a random function of Surface Material Category.
Image Capture Date and Time (NITF)	S	12	YYMMDDHHMMSS	The date and time of day that an SSDB image was captured, where YYMMDD = Year, Month and Day, and HHMMSS = Hours (0..24), Minutes and Seconds.
Image Capture Date Range (NITF)	S, S	13	YYMMDD, YYMMDD	Parameter indicating the preferred range of dates in which an image was captured for a texture.
Image Classification Authority (NITF)	S	20	--	The identity of the classification authority for an image. The code shall be in accordance with the regulations governing the appropriate security channels.

TABLE A-III

Fieldname	Type	Length	Range	Description
Image Codewords (NITF)	S	40	--	Security compartments associated with an image. Digraphs in accordance with DIAM 65-19 and its supplements, trigraphs not contained in DIAM 65-19, and complete codewords or project numbers may be used. The selection of a relevant set of codewords is implementation specific. The individual values are separated by single spaces.
Image Comment Field n (NITF)	S	80	--	Field to be used for free form comments. May be used for image specific information. If the comment is classified, then it will be preceded by the classification, including codeword(s). Omitted if Number of Image Comments is zero.
Image Compression Field (NITF)	S	2	"NC", "C0, "C1", "C2"	If the image is transmitted in a compressed form, the letter C followed by a number between 0 and 2 is used to indicate the compression scheme used (C0 = compressed with a user specified algorithm, C1 = one bit, C2 = ARIDPCM). Given as NC if the image is not compressed.
Image Control & Handling (NITF)	S	40	--	Security handling instructions associated with an image.
Image Coordinate System (NITF)	S	1	"G", "N"	Coordinate system of the image where G = geodetic, N = None. While NITF allows other values, P2851 has restricted the range of this field.

TABLE A-III

Fieldname	Type	Length	Range	Description
Image Coordinates (NITF)	INT2D	11	0..99999, 0..99999	X and Y location within an image.
Image Date & Time (NITF)	S	14	DDHHMMSSZMONYY	Time (Zulu) of acquisition of the image where DD is the day of the month, HH is the hour, MM is the minute, SS is the second, the character Z, MON is the first three characters of the month, and YY is the year.
Image Downgrading Event (NITF)	S	40	--	If the Image Security Downgrade equals "999998" then this field must be present and must specify the event.
Image File Creation Date and Time (NITF)	S	12	YYMMDDHHMMSS	The date and time of day that an SSDB image was created, where YYMMDD = Year, Month and Day, and HHMMSS = Hours (0..24), Minutes and Seconds.
Image Filter Condition Field (NITF)	S	1	"N"	Other values are reserved for future use.

TABLE A-III

Fieldname	Type	Length	Range	Description
Image Geographic Location (NITF)	S	91	DDMMSSSSSXb DDDMMSSSSSY (4 times)	Geographic location of the image in geodetic, UTM, or geocentric coordinates. Geodetic coordinates are given as the latitude and longitude of the four corners in clockwise order beginning with the top left corner of the image as it is transmitted, where DDMMSSSSSX represents degrees, minutes, and thousandths of seconds of latitude with X= N or S for north or south, and DDDMMSSSSSY represents degrees, minutes, and thousandths of seconds of longitude with Y = E or W for east or west. For non-geodetic images, these coordinates are expressed in meters. b = blank (" "). Omitted if Image Coordinate System equals None.
Image ID (NITF)	S	10	--	Textual identification of the image; unique across entire GTDB.
Image Location (NITF)	S	10	RRRRRCCCCC	An ordered pair defining the location in cartesian coordinates where the first pixel of the first line of the image is to be located, where RRRRR is the row and CCCCC is the column where the upper left corner of the image is to be located.
Image Magnification (NITF)	S	4	--	The magnification (or reduction) factor of the transmitted image relative to the original source image.

TABLE A-III

Fieldname	Type	Length	Range	Description
Image Mode (NITF)	S	1	"S, "I"	Flag indicating band sequential ("S") or band interleaved ("I") transmission format.
Image Quality Comment (NITF)	S	80	--	Free text comment field to include information pertaining to the quality of the image.
Image Quality Rating (NITF)	E	9	EXCELLENT, GOOD, FAIR, POOR	Rating of the quality of the image based on clarity and content of the image
Image Releasing Instructions (NITF)	S	40	--	A list of countries and/or groups of countries to which the data are authorized for release.
Image Security Classification (NITF)	S	1	"T", "S","C", "R", "U"	Classification of the image and image sub-header. T = Top Secret, S = Secret, C = Confidential, R = Restricted, U = Unclassified.
Image Security Control Number (NITF)	S	20	--	Security control numbers associated with the image. The format is in accordance with the regulations governing the appropriate security channel(s).
Image Security Downgrade (NITF)	S	6	--	An indicator which designates the point in time at which a declassification or downgrading action is to take place.

TABLE A-III

Fieldname	Type	Length	Range	Description
Image Source (NITF)	S	80	--	Description of the source of the image. If the source is classified, then it will be preceded by the classification, including codeword(s).
Image Sync Code (NITF)	I	1	0, 4	A field that indicates whether a synchronization code has been provided for uncompressed or ARIDPCM compressed data.
Image Title (NITF)	S	80	--	Title of the image.
Image-to-Image Contrast Enhancement Flag (NITF)	B	5	TRUE, FALSE	Flag indicating whether contrast enhancements have been performed between images.
Image Type (NITF)	S	8	--	The type of image, such as BW for black and white, TV, SAR, XRAY, MS for multispectral, FAX for facsimile, or IR. Multispectral may be further denoted by TM7 for Thematic Mapper band 7.
Inner Image Contrast Enhancement Flag (NITF)	B	5	TRUE, FALSE	Flag indicating whether contrast enhancements have been performed within an image.
Internal Material Category	I	4	1..127	Category code for material internal to an object.
Internal Material Volume	R6	12	0.0..1.9342 e+25	Amount of material internal to an object, in liters.

TABLE A-III

Fieldname	Type	Length	Range	Description
Island Boundary Point Count	I	11	4..2147483647	The number of Boundary Point records used to define the boundaries of an island within the gaming area.
Island Count	I	11	1..2147483647	The number of islands of culture detail defined for a GTDB.
Island Number	I	11	1..2147483647	A sequence number used to uniquely identify an island within a GTDB.
Kappa (NITF)	R6	12	0.0..360.0	A rotation angle around the z-axis. A positive angle rotates the x-axis toward the y-axis (expressed in degrees).
Last Maintenance Date and Time (NITF)	S	12	YYMMDDHHMMSS	The last date and time of day that a GTDB image was modified, where YYMMDD = Year, Month and Day, and HHMMSS = Hours (0..24), Minutes and Seconds.
Last Update Date	S	6	YYMMDD	The date of the most recent subarea update to the GTDB. If no updates have occurred, the date is equal to the Compilation Date.

TABLE A-III

Fieldname	Type	Length	Range	Description
Lat/Long NE Corner	I2D	21	-324000000.. 324000000, -648000000.. 648000000	A geographic coordinate defining the northeast corner of a geographic rectangle, such as an areal photo texture map. The coordinate consists of a latitude and a longitude, separated by an ASCII null character. See field definitions for Latitude and Longitude.
Lat/Long NW Corner	I2D	21	-324000000.. 324000000, -648000000.. 648000000	A geographic coordinate defining the northwest corner of a geographic rectangle, such as an areal photo texture map. The coordinate consists of a latitude and a longitude, separated by an ASCII null character. See field definitions for Latitude and Longitude.
Lat/Long SE Corner	I2D	21	-324000000.. 324000000, -648000000.. 648000000	A geographic coordinate defining the southeast corner of a geographic rectangle, such as an areal photo texture map. The coordinate consists of a latitude and a longitude, separated by an ASCII null character. See field definitions for Latitude and Longitude.

TABLE A-III

Fieldname	Type	Length	Range	Description
Lat/Long SW Corner	I2D	21	-324000000.. 324000000, -648000000.. 648000000	A geographic coordinate defining the southwest corner of a geographic rectangle, such as an areal photo texture map. The coordinate consists of a latitude and a longitude, separated by an ASCII null character. See field definitions for Latitude and Longitude.
Latitude	I	10	+-324000000	Geographic latitude expressed in thousandths of seconds of arc above or below the equator.
Latitude Interval	E	18	ONE_METER, TEN_METERS, THIRTY_METERS, ONE_HUNDRED_METERS	In a TGAB, the nominal latitude interval used to space terrain elevation matrix positions, from a list of valid values. In a TPAB, the interval in the SSDB grid used to generate the terrain polygons.
Latitude/Longitude (NITF)	S	22	HDDMMSSSSSb HDDDMMSSSSS	Actual ground location that is being used for an image control point or a boundary point on an areal texture footprint, where H = hemisphere, DD or DDD = degrees and SSSSS = thousandths of seconds, and b = blank (" "). Location is in absolute coordinates.
Latitudinal Origin	R10	16	-90.0..90.0	A user-defined CDBTP parameter indicating the latitudinal origin to be applied during coordinate transformations.

TABLE A-III

Fieldname	Type	Length	Range	Description
Layer Number (Infrared)	I	11	0..2147483647	A relative priority number indicating the sequence in which overlapping culture features, model polygons, or textures should be rendered for infrared simulation. Higher values indicate higher priority.
Layer Number (Radar)	I	11	0..2147483647	A relative priority number indicating the sequence in which overlapping culture features, model polygons, or textures should be rendered for radar simulation. Higher values indicate higher priority.
Layer Number (Visual)	I	11	0..2147483647	A relative priority number indicating the sequence in which overlapping culture features, model polygons, or textures should be rendered for visual simulation. Higher values indicate higher priority.
Layer Number				Same as Layer Number (Visual).
Left Tile Neighbor ID (NITF)	I	10	0..2147483647	The identifier of the neighboring model specific image to the left of the current image.

TABLE A-III

Fieldname	Type	Length	Range	Description
Length	R10	16	-9.999999999E+99.. 9.999999999E+99	The long dimension of a point feature.
Length of 1st Image (NITF)	I	10	0..1073741824	Length in bytes of the first image. If the image is compressed, the length after compression is given. The value of zero indicates that the length of the first image is not currently known and should be computed from the decompressed sizes given in the image sub-header.
Length of 1st Image Sub-Header (NITF)	I	6	000000..111000	Length in bytes of the sub-header of the first image.
Length of Attribute	I	3	0..80	Indicates the size (number of character positions) of the field used to store the value of an attribute in a FACS Record.
Length of Nth Image (NITF)	I	10	0..1073741824	Length in bytes of the Nth image. If the image is compressed, the length after compression is given. The value of zero indicates that the length of the Nth image is not currently known and should be computed from the decompressed sizes given in the image sub-header. N is the image number.
Length of Nth Image Sub-Header (NITF)	I	6	000000..111000	Length in bytes of the sub-header of the Nth image. N is the image number.

TABLE A-III

Fieldname	Type	Length	Range	Description
Light Delta (Point Light String Delta)	R3D10	50	-9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99	Distance between lights in a point light string, expressed as a vector using relative coordinates. The coordinates are longitude and latitude, represented in thousandths of seconds when the coordinate system is GEODETIC_FLOAT, and otherwise in meters; and elevation, always represented in meters.
Light Horizontal Center	R10	16	0.0..359.9999999	Angular offset to center of light lobe, in degrees.
Light Horizontal Fall	R10	16	0.0..359.9999999	Angle at which light intensity falls off, in degrees.
Light Horizontal Width	R10	16	0.0..359.9999999	Half angle of lobe width from lobe center, in degrees.
Light Intensity	I	6	0..32767	Candlepower of light.
Light String Shape	E	11	LINEAR, CURVILINEAR	General shape of a point light string.
Light Type	E	11	RUNWAY, APPROACH, VASI, TAXI, BEACONS, STROBES, OBSTRUCTION, CULTURAL, AIRCRAFT, OTHER, NONE	Code indicating functional type for a point light or point light string feature.

TABLE A-III

Fieldname	Type	Length	Range	Description
Light Vertical Center	R10	16	0.0..359.9999999	Angular offset to center of light lobe, in degrees.
Light Vertical Fall	R10	16	0.0..359.9999999	Angle at which light intensity falls off, in degrees.
Light Vertical Width	R10	16	0.0..359.9999999	Half angle of lobe width from lobe center, in degrees.
Linear Feature Area Block Flag	B	1	T, F	Boolean flag indicating whether a Linear Feature Area Block (LFAB) pseudo-file exists within the given area block.
LL Corner X/Y Image Coordinates (NITF)	I2D	15	-999999..999999, -999999..999999	X/Y cartesian coordinates of the lower left corner of the image.
LOD Resolution Description	S	80	--	Textual description of a model LOD resolution (e.g., in meters, number of polygons, etc.).
Long Lineal	B	1	T, F	Boolean flag indicating whether a point feature may be portrayed as a linear feature when viewed by radar.
Longitude	I	10	-648000000.. 648000000	Geographic longitude expressed in thousandths of seconds of arc on either side of the Prime Meridian.
Longitude Interval	E	18	ONE_METER, TEN_METERS, THIRTY_METERS, ONE_HUNDRED_METERS	The interval of longitude used to space terrain elevation matrix positions in a TGAB. In a TPAB, the interval of the SSDB grid used to generate the terrain polygons.

TABLE A-III

Fieldname	Type	Length	Range	Description
Longitudinal Origin	R10	16	-9.999999999E+99.. 9.999999999E+99	A user-defined CDBTP parameter indicating the longitudinal origin to be applied during coordinate transformations.
Low Level Effects	B	1	T, F	When true, indicates normalcy to the terrain plate, and therefore is an indication of higher radar backscatter. Default values are generated synthetically by P2851 software as a function of Feature Descriptor Code.
Low Value Field	R	6	0.0..100.0	Percentage.
LR Corner X/Y Image Coordinates (NITF)	I2D	15	-999999..999999, -999999..999999	X/Y cartesian coordinates of the lower right corner of the image.
LUT Entry Data (NITF)	S	7	--	Value within a lookup table for SMC/FDC texture. The NITF limit of 1 byte is extended here for P2851 purposes to 7 bytes. This value uses ASCII alphanumeric characters rather than binary data. The first two characters represent the SMC code (00-15) while the next five characters represent the FDC value.
Match Terrain At SLODS Flag	B	1	T, F	True Value means that terrain polygons from adjacent area blocks in different SLODs merge smoothly by virtue of the fact that the vertices along the shared boundary are identical in both area blocks.

TABLE A-III

Fieldname	Type	Length	Range	Description
Maximum Areal Feature Polygons	I	11	0..2147483647	The largest number of areal feature polygons occurring in any area block within a SLOD.
Maximum Elevation	I	7	-120000..100000	The highest terrain elevation value contained within a TGAB.
Maximum Linear Feature Segments	I	11	0..2147483647	The largest number of linear feature segments occurring in any area block within a SLOD.
Maximum Model Polygons	I	6	0..32767	The largest number of referenced model polygons occurring in any a... block within a SLOD.
Maximum Model References	I	11	0..2147483647	The largest number of model references occurring in any area block within a SLOD.
Maximum Number of Culture Polygons	I	11	0..2147483647	User-defined CDBTP parameter indicating an upper limit on the number of areal feature polygons allowed in a given area block.
Maximum Number of Edges	I	6	0..20000	User-defined CDBTP parameter indicating an upper limit on the number of edges used to define an areal feature polygon.
Maximum Number of Model References	I	11	0..2147483647	User-defined CDBTP parameter indicating an upper limit on the number of model references allowed in a given area block.
Maximum Number of Model Polygon Edges	I	6	0..20000	The maximum number of edges allowed in a model polygon. 0 represents no limit.

TABLE A-III

Fieldname	Type	Length	Range	Description
Maximum Number of Terrain Polygons	I	11	0..2147483647	User-defined CDBTP parameter indicating an upper limit on the number of terrain polygons allowed in a given area block.
Maximum Point Features	I	11	0..2147483647	The largest number of point features occurring in any area block within a SLOD.
Maximum Point Light Features	I	11	0..2147483647	The largest number of point light features occurring in any area block within a SLOD.
Maximum Point Light Strings	I	11	0..2147483647	The largest number of point light strings occurring in any area block within a SLOD.
Maximum Terrain Polygons	I	11	0..2147483647	The largest number of terrain polygons occurring in any area block within a SLOD.
Maximum Texture References	I	11	0..2147483647	The largest number of texture references occurring in any area block within a SLOD.
Maximum Total Elements	I	11	0..2147483647	The largest number of total data base elements occurring in any area block within a SLOD. Elements counted include areal feature polygons, linear feature segments, point features, point lights, point light strings, model references, model polygons, terrain polygons, and photo texture references.

TABLE A-III

Fieldname	Type	Length	Range	Description
Mean Square Error Lat/Lon/ Height (NITF)	S, S, R10		DDDMMSSSSS, DDDMMSSSSS, 0.0.. 9.999999999E+99	A measure of accuracy which includes both bias squared and variance. *DDD* = degrees, *MM* = minutes, SSSSS = thousandths of seconds, and the REAL10 value is in meters squared.
Mean Square Error Omega/Phi/ Kappa (NITF)	S, S, R10, S		DDDMMSSSSS, DDDMMSSSSS, 0.0.. 9.999999999E+99, DDDMMSSSSS	A measure of accuracy which includes both bias squared and variance; DDD = degrees, MM = minutes, SSSSS = thousandths of seconds, and the REAL10 value is in meters squared.
Measured Image Coordinates (NITF)	R6	12	-1.0..1.0	The position of the fiducial marks as measured by a comparator on a particular photograph (expressed in meters).
Message Classification Authority (NITF)	S	20	--	The identification of the classification authority for the message. The code shall be in accordance with the regulations governing the appropriate security *channel(s)*.
Message Codewords (NITF)	S	40	--	Security compartments associated with the message.
Message Control & Handling (NITF)	S	40	--	Security handling instructions associated with the message.
Message Copy Number (NITF)	I	5	0..99999	Copy number of the message.

TABLE A-III

Fieldname	Type	Length	Range	Description
Message Date & Time (NITF)	S	1.	DDHHMMSSZMONYY	Time (Zulu) of of origination of the message, where DD is the day of the month, HH is the hour, MM is the minute, SS is the second, the character Z, MON is the first three characters of the month, and YY is the year.
Message Downgrading Event (NITF)	S	40	--	If the Message Security Downgrade equals "999998" then this field must be present and must specify the event.
Message Length (NITF)	I	12	0..999999999999	The length in bytes of the entire message including all headers, sub-headers and data.
Message Number of Copies (NITF)	I	5	0..99999	Total number of copies of the message.
Message Part Type (NITF)	S	2	"IM"	Given as "IM" to identify the sub-header as an image sub-header.
Message Releasing Instructions (NITF)	S	40	--	A list of countries and/or groups of countries to which the data are authorized for release.
Message Security Classification (NITF)	S	1	"T", "S", "C", "R", "U"	Classification of the entire message, where T = Top Secret, S = Secret, C = Confidential, R = Restricted, U = Unclassified.
Message Security Control Number (NITF)	S	20	--	Security control number associated with the message.

TABLE A-III

Fieldname	Type	Length	Range	Description
Message Security Downgrade (NITF)	S	6	--	An indicator which designates the point in time at which a declassification or downgrading action is to take place.
Message Title (NITF)	S	80	--	Title of message.
Message Type & Version (NITF)	S	9	NITFNN.NN	A character string which indicates this message is using version NN.NN of NITF.
Microdescriptor Type	E	20	HOMOGENEOUS_AREA, DRAINAGE, PATTERN_DISTRIBUTION, TRANSPORTATION, VEGETATION, WEATHER_EFFECTS, SEASONAL_EFFECTS, TIME_OF_DAY, GROUND_CONDITIONS, ALTERNATE_ATTRIBUTES, VERTICALLY_COMPOSITE, TEMPORAL_EFFECTS	Designates the specific micro-descriptor and attribute being described, from a standard list of microdescriptor formats.
Microdescriptor Value(s)	S	80	--	One or more data values associated with a given microdescriptor attribute. When more than one value is required, they will be separated by ASCII null characters.
Minimum Areal Feature Polygons	I	11	0..2147483647	The smallest number of areal feature polygons occurring in any area block within a SLOD.
Minimum Elevation	I	7	-120000..100000	The lowest terrain elevation value contained within a TGAB.

TABLE A-III

Fieldname	Type	Length	Range	Description
Minimum Linear Feature Segments	I	11	0..2147483647	The smallest number of linear feature segments occurring in any area block within a SLOD.
Minimum Model Polygons	I	6	0..32767	The smallest number of referenced model polygons occurring in any area block within a SLOD.
Minimum Model References	I	11	0..2147483647	The smallest number of model references occurring in any area block within a SLOD.
Minimum Number of Terrain Polygons	I	11	0..2147483647	User-defined CDBTP parameter indicating a lower limit on the number of terrain polygons generated in a given area block.
Minimum Point Features	I	11	0..2147483647	The smallest number of point features occurring in any area block within a SLOD.
Minimum Point Light Features	I	11	0..2147483647	The smallest number of point light features occurring in any area block within a SLOD.
Minimum Point Light Strings	I	11	0..2147483647	The smallest number of point light strings occurring in any area block within a SLOD.
Minimum Terrain Polygons	I	11	0..2147483647	The smallest number of terrain polygons occurring in any area block within a SLOD.
Minimum Texture References	I	11	0..2147483647	The smallest number of texture references occurring in any area block within a SLOD.

TABLE A-III

Fieldname	Type	Length	Range	Description
Minimum Total Elements	I	11	0..2147483647	The smallest number of total data base elements occurring in any area block within a SLOD. Elements counted include areal feature polygons, linear feature segments, point features, point lights, point light strings, model references, model polygons, terrain polygons, and photo texture references.
Mirror	B, B, B, B	7	T, F; T, F; T, F; T, F	Flag indicating whether a texture map can be mirrored along the left, right, top, and bottom edges.
Model Description	S	80	--	Description of a model.
Model Library Type (NITF, non-NITF)	E	15	TWO_D_STATIC, THREE_D_STATIC, THREE_D_DYNAMIC	Distinguishes among the three types of P2851 model libraries -- 2-D Static, 3-D Static, and 3-D Dynamic.
Model List Count	I	11	0..2147483647	The number of models specifically requested by the user for inclusion in the GTDB.
Model LOD	E	5	LOD_0, LOD_1, LOD_2, LOD_3, LOD_4, LOD_5, LOD_6, LOD_7, LOD_8	Designates the level of detail category for a given version of a given model.
Model Name (NITF, non-NITF)	S	65	--	A descriptive textual name for a model.
Model Number (Non-NITF)	I	11	1..99999	A number which uniquely identifies a model within a P2851 model library.
Model Number (NITF)	I	10	1..99999	A number which uniquely identifies a model within a P2851 model library.

TABLE A-III

Fieldname	Type	Length	Range	Description
Model Reference Area Block Flag	B	1	T, F	Boolean flag indicating whether a Model Reference Area Block (MRAB) pseudo-file exists within the given area block.
Model Reference Number	I	11	0..2147483647	The Model Number of a model referenced by a terrain polygon.
Model Reference Point	R2D6	25	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	The location of a point on a 2D model that is used to reference model-based texture application; corresponds to the texture origin.
	R3D6	38	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	The location of a point on a 3D model that is used to reference model-based texture application; corresponds to the texture origin.
Model Tie Point ID (NITF)	I	10	0..2147483647	A unique identifier of a model tie point.
Model View Description (NITF)	S	60	--	Textual description of the view cf a model presented within an image, i.e., "Right Side of Truck".
Model-Based Mapping Flag	B	1	T, F	Parameter indicating the existence of model-based mapping parameters for texture.
Modified Specific Texture Flag (NITF)	B	5	TRUE, FALSE	Flag indicating whether a specific texture has been modified with synthetic data.

TABLE A-III

Fieldname	Type	Length	Range	Description
Monitor Type	E	12	NO_MONITOR, LONGITUDINAL, TRANSVERSE, MODIFIED	Describes raised portion of roof.
NITF Header Length (NITF)	I	6	000000..276380	The length in bytes of the NITF header.
Noise Removal Flag (NITF)	B	5	TRUE, FALSE	Flag indicating whether noise removal operations have been performed on the image.
Normal List Position	I	11	1..2147483647	An index into a coordinate list, identifying a vector used as a normal to a polygon or vertex.
North Tile Neighbor ID (NITF)	I	10	0..2147483647	The identifier of the neighboring areal specific image to the north.
Number of Area Blocks	I	11	1..2147483647	The number of area blocks within a SLOD.
Number of Areal Feature Polygons	I	11	0..2147483647	The number of areal feature polygons occurring within an area block.
Number of Audio Segments (NITF)	I	3	000	Not currently supported within NITF, therefore, this value is always 0.

TABLE A-III

Fieldname	Type	Length	Range	Description
Number of Bands (NITF)	I	1	1..9	The number of bands of image data in the message. Used for color imagery, pseudocolor or multispectral images. The sequence of bands shall be determined by examining the Band Image Type Field. For single band images, the Number of Bands shall be 1.
Number of Bits Per Pixel Per Band (NITF)	I	2	01..64	The number of data bits for each texel for each band in the image before compression. For multi-band images treated as a single image, the number of bits per texel is identical for each band. Standard values include 1, 8, and 16 for intensity, color, or other multispectral textures; 56 for SMC/FDC textures, and 16 or 24 for terrain. Same as Bits Per Texel Per Band for non-NITF data.
Number of Blocks Per Column (NITF)	I	4	1..9999	The number of image blocks in a column in the vertical direction. (P2851 has relaxed the NITF restriction of one block per column).
Number of Blocks Per Row (NITF)	I	4	1..9999	The number of image blocks in a row or line in the horizontal direction. (P2851 has relaxed the NITF restriction of one block per row).
Number of Boundaries (NITF)	I	5	0..32767	The total number of outlines required to specify the coverage or "footprint" of the area being transmitted.

TABLE A-III

Fieldname	Type	Length	Range	Description
Number of Boundary Points (NITF)	I	5	0..32767	The total number of coordinates that define an outline around an area of data that is being transmitted.
Number of Collision Test Points	I	4	0..255	The number of collision test point vertices defined for a given model.
Number of Color References	I	11	1..2147483647	The number of occurrences of a given color within a given unit of the data base.
Number of Component Texture Reference Pointers	I	6	0..65535	Number of textures referenced by a model component at the component level.
Number of Components	I	6	0..1000	Number of components in a model.
Number of Control Points (NITF)	I	5	0..32767	The total number of control points.
Number of Culture Color Tables	I	11	0..2147483647	The number of Culture Color Table records associated with a model library, *SLOD*, or area block. Each record contains a single entry in a distribution table of colors used in models and features.
Number of Culture References	I	11	0..2147483647	The number of culture features lying on a given terrain polygon.
Number of Culture Vertices	I	11	0..2147483647	The number of coordinate vertices used to define culture features within a given area block.

TABLE A-III

Fieldname	Type	Length	Range	Description
Number of Data Files	I	6	2..32767	The number of data files used for a given texture. All Stage 1 textures have 2 or more data files; all other textures have exactly two data files: the NITF Image Sub-Header File and the NITF Image Data File.
Number of Data Source Table Entries (NITF)	I	4	0..1000	Number of data sources in a data source table.
Number of Delete-List Entries	I	11	0..2147483647	The number of Delete-List records associated with a SLOD. Each record contains a user-defined range of feature FDCs which should be omitted from the GTDB.
Number of Face-Based Texture References	I	11	0..2147483647	The number of texture references using the face-based texture mapping method. Used by models and areal features in area blocks.
Number of FACS	I	4	0..255	The number of FACS records associated with a given feature.
Number of FACS Table Entries	I	11	0..2147483647	Number of entries in a FACS table.
Number of Feature Distribution Tables	I	11	0..2147483647	The number of Feature Distribution Table records associated with a SLOD or area block. Each record contains a single entry in a distribution table of Feature Descriptor Codes.

TABLE A-III

Fieldname	Type	Length	Range	Description
Number of Feature Occurrences	I	11	0..2147483647	The number of occurrences of features with a given Feature Descriptor Code within a given unit of the GTDB.
Number of Features	I	11	0..2147483647	The number of features of a given type occurring in a given area block.
Number of Fiducial Coordinates (NITF)	I	5	0..32767	Total number of fiducial coordinates associated with an image.
Number of Fragments	I	11	0..2147483647	The number of features of a given type occurring in a given area block, after features have been fragmented along underlying terrain boundaries. Each fragment is counted individually.
Number of Generic Texture Sets (NITF)	I	10	0..2147483647	The total number of generic texture sets in a texture transmittal, where a generic texture set is a set of generic textures that represent the same entity, and each member of the set has a different size and/or resolution.
Number of Generic Textures in Set (NITF)	I	10	0..2147483647	The number of generic textures included within a generic texture set.
Number of Geographic Tie Points (NITF)	I	10	0..2147483647	The total number of tie points for an areal texture map.

TABLE A-III

Fieldname	Type	Length	Range	Description
Number of Global-Based Texture References	I	11	0..2147483647	The number of texture references using the global-based texture mapping method. Used by areal features in area blocks.
Number of Horizontal Blocks Field	I	11	0..2147483647	The number of horizontal blocks in a texture.
Number of Image Comments (NITF)	I	1	0..9	The number of free form image comments.
Number of Images (NITF)	I	3	000..999	The number of separate images included in the message.
Number of Keep-List Entries	I	11	0..2147483647	The number of Keep-List records associated with a SLOD. Each record contains a user-defined range of feature FDCs which must be included in the GTDB.
Number of Labels (NITF)	I	3	000..999	The number of separate labels included in the message.
Number of Layers Above Terrain Polygon	I	11	0..2147483647	The maximum number of layers of overlapping culture features over a given terrain polygon.
Number of Level-List Entries	I	11	0..2147483647	The number of Level-List records associated with a SLOD. Each record contains a user-defined range of feature FDCs under which the terrain must be level in the GTDB.

TABLE A-III

Fieldname	Type	Length	Range	Description
Number of Light Color Tables	I	11	0..2147483647	The number of Light Color Table records associated with a model library, SLOD, or area block. Each record contains a single entry in a distribution table of colors used in point light features.
Number of Lights	I	4	1..255	The number of individual lights in a point light string.
Number of Linear Feature Segments	I	11	0..2147483647	The number of linear feature line segments occurring within an area block.
Number of LOD Texture Reference Pointers	I	6	0..65535	Number of textures referenced by a model LOD at the LOD level.
Number of LODs	I	2	1..9	The number of versions of a given model in a model library, where the different versions represent different levels of detail.
Number of LUT Entries (NITF)	I	5	00001..65536	The number of entries in each of the look-up tables for a band of an image.
Number of LUTs (NITF)	I	1	0..4	The number of look-up tables used in displaying a band of an image.
Number of Mapped Texture Reference Pointers	I	11	0..2147483647	The number of texture references associated with a cultural areal feature for texture that has been mapped to that feature.

TABLE A-III

Fieldname	Type	Length	Range	Description
Number of Microdescriptors	I	4	0..255	The number of microdescriptor records associated with a given feature, model, or model polygon.
Number of Model-Based Texture References	I	11	0..2147483647	The number of texture references using the model-based texture mapping method. Used by models.
Number of Model FACS	I	4	0..255	The number of Model FACS records associated with a given model.
Number of Model LODs	I	2	1..9	The number of versions of a given model available in the model library, where each version represents a different level of detail.
Number of Model References	I	11	0..2147483647	The number of model reference records occurring within an area block.
Number of Model Tie Points (NITF)	I	10	0..2147483647	The total number of tie points for a model texture map.
Number of Models	I	11	0..2147483647	The number of models, not counting different LODs of the models, in a model library.
Number of Models in Image (NITF)	I	3	0..999	The number of models that are represented in some manner within an image.
Number of Non-Mapped Texture References	I	11	0..2147483647	The number of texture references associated with a culture feature or a model LOD for texture that has not been mapped.

TABLE A-III

Fieldname	Type	Length	Range	Description
Number of Non-Static Presentations (NITF)	I	3	000	The number of non-static presentation information files included in the message.
Number of Occurrences	I	11	0..2147483647	The number of occurrences of a given data condition within a given unit of the GTDB.
Number of Pixels Per Block Horizontal (NITF)	I	5	00001..99999	The number of pixels horizontally in each block.
Number of Pixels Per Block Vertical (NITF)	I	5	00001..99999	The number of pixels vertically in each block.
Number of Point Features	I	11	0..2147483647	The number of point features occurring within an area block.
Number of Point Light Features	I	11	0..2147483647	The number of point light features occurring within an area block.
Number of Point Light Strings	I	11	0..2147483647	The number of point light strings occurring within an area block.
Number of Polygon FACS	I	4	0..255	The number of Polygon FACS records associated with a given polygon within a given model.
Number of Polygon Texture Reference Pointers	I	6	0..65535	Number of textures referenced by a polygon in a model.
Number of Polygons	I	6	0..32767	The number of polygons making up a given LOD of a given model.
Number of Sensors (NITF)	I	3	0..127	Number of sensors used to form a composite processed image.

TABLE A-III

Fieldname	Type	Length	Range	Description
Number of Separation Planes	I	6	0..32767	The number of Separation Plane records associated with a given model.
Number of SLODs	I	11	1..2147483647	The number of Simulator Levels of Detail defined for terrain and culture data within the GTDB. This is a user-defined value.
Number of SMC Distribution Tables	I	11	0..2147483647	The number of SMC Distribution Table records associated with a SLOD or area block. Each record contains a single entry in a distribution table of Surface Material Categories.
Number of Stage 1 Specific Textures	I	11	0..2147483647	The total number of Stage 1 Specific Texture maps included within a GTDB database.
Number of Stage 1 Texture File Associations	I	11	0..2147483647	The total number of Stage 1 Texture files (originating from another source) where the file name from the original source has been associated with a new GTDB file name included within a GTDB database. Every Stage 1 original source data file shall have such an association.
Number of Stage 2 Specific Textures	I	11	0..2147483647	The total number of Stage 2 Specific Texture maps included within a GTDB database.
Number of Stage 3 Generic Textures	I	11	0..2147483647	The total number of Stage 3 Generic Texture maps included within a GTDB database.

TABLE A-III

Fieldname	Type	Length	Range	Description
Number of Stage 3 SMC/FDC Textures	I	11	0..2147483647	The total number of Stage 3 SMC/FDC Texture maps included within a GTDB database.
Number of Stage 3 Specific Textures	I	11	0..2147483647	The total number of Stage 3 Specific Texture maps included within a GTDB database.
Number of Stage 4 Generic Textures	I	11	0..2147483647	The total number of Stage 4 Generic Texture maps included within a GTDB database.
Number of Stage 4 SMC/FDC Textures	I	11	0..2147483647	The total number of Stage 4 SMC/FDC Texture maps included within a GTDB database.
Number of Stage 4 Specific Textures	I	11	0..2147483647	The total number of Stage 4 Specific Texture maps included within a GTDB database.
Number of Stage 5 Generic Textures	I	11	0..2147483647	The total number of Stage 5 Generic Texture maps included within a GTDB database.
Number of Stage 5 SMC/FDC Textures	I	11	0..2147483647	The total number of Stage 5 SMC/FDC Texture maps included within a GTDB database.
Number of Stage 5 Specific Textures	I	11	0..2147483647	The total number of Stage 5 Specific Texture maps included within a GTDB database.
Number of Stereo Mates (NITF)	I	3	0..127	Number of images that overlap a given image.

TABLE A-III

Fieldname	Type	Length	Range	Description
Number of Structures	I	11	0..2147483647	Approximate number of structures per square kilometer contained within an areal feature. Synthetic default values are generated for non-DMA data by P2851 software as a random function of Feature Descriptor Code.
Number of Subsidiary Model References	I	11	0..2147483647	The number of Subsidiary Model Reference records associated with a given model.
Number of Symbols (NITF)	I	3	000..999	The number of separate symbols included in the message.
Number of Terrain Polygons	I	11	0..2147483647	The number of terrain polygons generated within an area block.
Number of Text Files (NITF)	I	3	000..999	The number of separate text files that are included in the message.
Number of Texture Images in Library	I	11	0..2147483647	The number of photo texture image maps within a photo texture library.
Number of Texture Pattern Coordinates	I	6	0..65535	Total number of coordinates associated with an image used to warp the image onto a model using vertex to vertex mapping techniques.
Number of Texture References	I	11	0..2147483647	Number of entries in a texture reference table. Also, the number of textures associated with a model polygon or a culture feature; the number of texture references for all features within an area block.

TABLE A-III

Fieldname	Type	Length	Range	Description
Number of Tie Point References (NITF)	I	10	0..2147483647	The number of texture maps that share a specific tie point.
Number of Vertex-to-Vertex Texture References	I	6	0..32767	The number of texture references using the vertex-to-vertex texture mapping method. Used by model polygons and terrain polygons.
Number of Vertical Blocks Field	I	11	0..2147483647	The number of vertical blocks in a texture.
Number of Vertices	I	11	0..2147483647	The number of unique coordinates defined to represent culture and polygonized terrain within an area block.
Number of Vertices (Model Polygon Record)	I	6	0..65534	The number of unique coordinates defined to represent a model polygon.
Number of Z-Density Distribution Tables	I	11	0..2147483647	The number of Z-Density Distribution Table records associated with a SLOD. Each record contains a single entry in a distribution table of levels of culture layering.
Object Or Material Texture Flag (NITF)	E	8	OBJECT, MATERIAL	Flag indicating whether a generic texture is applied to a certain object or if it is representative of a material (e.g., generic texture for a certain tree or a road would be classified as OBJECT; texture for tree bark or asphalt would be classified as MATERIAL).

TABLE A-III

Fieldname	Type	Length	Range	Description
Object Volume	R6	12	0.0..1.9342 e+25	The internal volume of an object, in liters.
Occlusion Removal Flag (NITF)	B	5	TRUE, FALSE	A flag indicating whether occluding objects have been removed from an image.
Offset Vector	R3D10	50	-9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99	The coordinates defining the position of a model within an area block. The name of this field is somewhat misleading, as the location is expressed in absolute rather than relative coordinates. The coordinates are longitude and latitude, represented in thousandths of seconds when the coordinate system is GEODETIC_FLOAT, and in meters otherwise; and elevation, always represented in meters.
Omega (NITF)	R6	12	0.0..360.0	A rotation angle around the x-axis. A positive angle rotates the y-axis toward the z-axis (expressed in degrees).
Orientation	R10	16	0.0..359.9999999	Orientation of a feature relative to north, in degrees. Synthetic default values are generated by P2851 software using a random function.
Orientation Angle	R10	16	0.0..359.9999999	Orientation of a model reference relative to north, in degrees.

TABLE A-III

Fieldname	Type	Length	Range	Description
Orientation Vectors	R2D10, R2D10	67	-9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99; -9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99	For 2D models or 2D culture, two vectors aligning the x and the y axes of a texture to be mapped with 2D model space or world space; used for mapping onto a model using model-based texturing or onto culture using global-based texturing.
	R3D10, R3D10	101	-9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99; -9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99	For 3D models, 3D culture, or 3D terrain polygons, two vectors aligning the x and the y axes of a texture to be mapped with 3D model space or world space; used for mapping onto a model using model-based texturing or onto terrain/culture using global-based texturing.
Origin of Eastings	R10	16	-9.999999999E+99.. 9.999999999E+99	A user-defined CDBTP parameter indicating the origin of eastings to be applied during coordinate transformations.
Origin of Northings	R10	16	-9.999999999E+99.. 9.999999999E+99	A user-defined CDBTP parameter indicating the origin of northings to be applied during coordinate transformations.
Original File Name	S	80	--	The original source's data file name for an image file stored on a GTDB tape as texture.

TABLE A-III

Fieldname	Type	Length	Range	Description
Originating Station ID (NITF)	S	10	--	Identification code of the originating system (terrain or texture).
Originator's Name (NITF)	S	27	--	Name of the operator who originated the message.
Originator's Phone Number (NITF)	S	18	--	Phone number of the operator who originated the message.
P2851 GTDB Catalog ID	S	7	Gnnnnnn	The unique identifier of a GTDB within the P2851 system. The letter G is followed by a 6-digit sequence number.
Percent of Cloud Cover (NITF)	I	3	0..100	Percentage of the image which is covered by clouds.
Percent of Roof Coverage	R6	12	0.0..100.0	Percentage of the surface of an areal feature consisting of roofs. Synthetic default values are generated by P2851 software as a random function of Feature Descriptor Code.
Percent of Shadow Cover (NITF)	I	3	0..100	Percentage of the image which is covered by shadows.
Percent of Specific Texture (NITF)	I	3	0..100	Percentage of an image that is specific to the geographic location that it is being placed at.
Percent of Texture in Tile (NITF)	I	3	0..100	Percentage of a Stage 3 texture tile that has been filled with actual texture (i.e., some void areas may exist within a texture tile.

TABLE A-III

Fieldname	Type	Length	Range	Description
Percent of Tree Coverage	R6	12	0.0..100.0	Percentage of the surface of an areal feature consisting of trees. Synthetic default values are generated by P2851 software as a random function of Feature Descriptor Code.
Percentage of Texture Coverage	R6	12	0.0..100.0	The percentage of texture covering a model LOD.
Phi (NITF)	R6	12	0.0..360.0	A rotation angle around the y-axis. A positive angle rotates the z-axis toward the x-axis (expressed in degrees).
Placement Point	R2D6	25	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	Point on a 2D model used for easy placement of the model.
	R3D6	38	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	Point on a 3D model used for easy placement of the model.
Point Feature Area Block Flag	B	1	T, F	Boolean flag indicating whether a Point Feature Area Block (PFAB) pseudo-file exists within the given area block.

TABLE A-III

Fieldname	Type	Length	Range	Description
Point Light Feature Area Flag	B	1	T, F	Boolean flag indicating whether a Block Point Light Feature Area Block (PLFAB) pseudo-file exists within the given area block.
Point Light String Feature Block Flag	B	1	T, F	Boolean flag indicating whether a Area Point Light String Feature Area Block (PLSFAB) pseudo-file exists within the given area block.
Polygon Alignment Vector	R2D6	25	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	Vector lying on the plane of the polygon and aligned in the direction of the x-axis of the texture map before the Rotation About Texture Origin value is applied; dimension is determined by the dimension of the polygon.
	R3D6	38	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	
Polygon ID	I	11	1..2147483647	Unique identifying number assigned to each polygon in a model.
Polygon Illumination Type	E	19	SELF_LUMINOUS, SUN_ILLUMINATION, NO_SUN_ILLUMINATION	Indicates how to calculate illumination effects for a feature or model polygon. Synthetic default values are generated by P2851 software as a function of Feature Descriptor Code.
Polygon Landing Light Illumination	B	1	T, F	Indicates whether a model polygon are illuminated by landing lights.

TABLE A-III

Fieldname	Type	Length	Range	Description
Polygon Long Dimension	R2D6	25	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	A coordinate representing the vector giving the longest distance between polygon vertices.
	R3D6	38	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	
Polygon Non-Occulting	B	1	T, F	When true, indicates that the color of a model polygon is additive with the background.
Polygon Non-Shadow	I	6	0..32767	The amount of shadow a model polygon presents when illuminated or irradiated.
Polygon Normal	R3D10	50	-9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99	A coordinate representing the vector perpendicular to the surface of a terrain or culture polygon.
	R3D6	38	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	A coordinate representing the vector perpendicular to the surface of a model polygon.

TABLE A-III

Fieldname	Type	Length	Range	Description
Polygon Reference Point	R2D6	25	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	A point on a 2D model polygon which corresponds to the origin of the texture being mapped.
	R3D6	38	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	A point on a 3D model polygon which corresponds to the origin of the texture being mapped.
	R2D10	33	-9.999999999E+99.. 9.999999999E+99, -9.999999999E+99.. 9.999999999E+99	A point on a 2D or 3D culture polygon or a 3D terrain polygon which corresponds to the origin of the texture being mapped. The point is given in relative longitude/ latitude coordinates, represented in thousandths of seconds when the coordinate system is *GEODETIC_FLOAT*, and in meters otherwise.
Polygon Short Dimension	R2D6	25	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	A coordinate representing the vector giving the shortest distance between polygon vertices.
	R3D6	38	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	

TABLE A-III

Fieldname	Type	Length	Range	Description
Position	R2D6	25	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	A coordinate representing the position of a light in a point light string in a model.
	R3D6	38	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	
Positional Accuracy Standards (NITF)	S	80	--	Description of standard used in positioning texture on geographic areas or models.
Predominant Height	R10	16	-9.999999999E+99.. 9.999999999E+99	Height of a feature above the terrain surface, at its longest vertical axis. Synthetic default values are generated by P2851 software as a random function of Feature Descriptor Code. (meters)
Processing Stage (NITF, non-NITF)	E	3	I, II, III, IV, V	Level of processing performed on a texture. (Stage I = original, raw image with control points and sensor information, if available; Stage II = image with noise removal, contrast enhancement, occlusion removal; Stage III = geometrically corrected, mosaicked textures; Stage IV = Stage III texture in projection other than geodetic (areal) or local cartesian (model); Stage V = Stage III or IV texture mapped onto polygons.)

TABLE A-III

Fieldname	Type	Length	Range	Description
Radius	R10	16	0.0.. 9.999999999E+99	Radius of curvature for a point light string; also, the radius of a circumscribing circle for a feature, model, or model polygon. (meters)
Rectification (NITF)	S	20	"RECTIFIED", "EPIPOLAR", "NONE", etc.	Definition of the type of rectification process used on an image.
Referenced Model Library Type	E	15	TWO_D_STATIC, THREE_D_STATIC, THREE_D_DYNAMIC	The model library that a referenced model belongs to.
Referenced Model LOD	E	5	LOD_0, LOD_1, LOD_2, LOD_3, LOD_4, LOD_5, LOD_6, LOD_7, LOD_8	The LOD of a model being referenced as a subsidiary model by another model.
Referenced Model Number	I	11	1..99999	The ID number of a model being referenced as a subsidiary model by another model.
Referencing Model LOD	E	5	LOD_0, LOD_1, LOD_2, LOD_3, LOD_4, LOD_5, LOD_6, LOD_7, LOD_8	The LOD of a model that is referencing another model as a subsidiary model.
Referencing Model Number	I	11	1..99999	The ID number of a model that is referencing another model as a subsidiary model.
Reflectance	R6	12	0.0..1.0	Ratio of radiant energy reflected by a feature or model, to the energy incident upon it. Synthetic default values are generated by P2851 software as a random function of Surface Material Category.

TABLE A-III

Fieldname	Type	Length	Range	Description
Relative Coordinates (NITF)	R3D6		-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	Model coordinates (x,y,z) used as control points or texture footprint boundary points for tying a texture to a model; each coordinate corresponds to an Image Coordinate; if the model is 2D, then z = 0.0.
Relative Latitude	I	11	-324000000.. 324000000	Geographic latitude expressed in thousandths of seconds of arc relative to a reference latitude.
Relative Longitude	I	11	-648000000.. 648000000	Geographic longitude expressed in thousandths of seconds of arc relative to a reference longitude.
Reliability of Data (NITF)	I	3	0..100	The degree of reliability of the data.
Right Tile Neighbor ID (NITF)	I	10	0..2147483647	The identifier of the neighboring model specific image to the right of the current image.
Roof Type	E	8	FLAT, SHED, DECK, GABLE, HIPPED, GAMBREL, MANSARD, SAWTOOTH, CURVED, CONICAL, NONE	Indicates shape of roof.

TABLE A-III

Fieldname	Type	Length	Range	Description
Rotation	R3D6	38	0.0..360.0, 0.0..360.0, 0.0..360.0	A set of vectors indicating how a model should be rotated in 3-D space when placed on a specific model reference location. The internal format of this field varies with the user-specified coordinate system for the GTDB.
Rotation About Texture Origin	R6	12	0.0..360.0	Rotation angle of a texture when applying it to a polygon; rotation is applied after the texture has been placed on the polygon using the other mapping parameters.
Rotation Angles	R3D6	38	0.0..360.0, 0.0..360.0, 0.0..360.0	Rotation angles about a subsidiary model's x, y, and z axes in its local coordinate system (right-handed).
Row Count	I	7	1..108000	The number of latitude "rows" of terrain elevation values contained within a Terrain Column record of a TGAB.
Scale Factor	R10	16	0.0.. 9.999999999E+99	A user-defined CDBTP parameter indicating the scale factor to be applied during coordinate transformations.
Scale Factor (Model References)	R3D10	50	0.0.. 9.999999999E+99, 0.0.. 9.999999999E+99, 0.0.. 9.999999999E+99	A set of three scale factors to be applied to the three axes of a model for a given instance of that model.

TABLE A-III

Fieldname	Type	Length	Range	Description
Scale Factor (Subsidiary Model Reference)	R2D6	25	0.0..1.93428E+25, 0.0..1.93428E+25	A set of 2 for 2-D, or 3 for 3-D scale factors to be applied to the axes of a subsidiary model for a given instance of that subsidiary model.
	R3D6	38	0.0..1.93428E+25, 0.0..1.93428E+25, 0.0..1.93428E+25	
Scanner Filter ID (NITF)	E	5	MONO, RED, GREEN, BLUE	Filter used when scanning a hardcopy image.
Scanner ID (NITF)	S	20	--	Identifier or name of the device used to scan an image.
Scanner Resolution (NITF)	S	10	--	The distance which is represented by a pixel in image space dimensions. The range is 0.1 micron to 1 mm in image dimensions.
Second Standard Parallel	R10	16	-90.0..90.0	A user-defined CDBTP parameter indicating a second standard parallel to be applied during projection transformations.
Security Level	S	2	--	The security classification for the GTDB or a file within the GTDB.
Self-Emitter	B	1	T, F	Flag indicating whether a feature or model polygon has self-heating characteristics. Synthetic default values are generated by P2851 software as a function of Feature Descriptor Code.

TABLE A-III

Fieldname	Type	Length	Range	Description
Semi-Major Axis	R10	16	-9.999999999E+99.. 9.999999999E+99	A user-defined CDBTP parameter indicating the semi-major axis to be employed during projection transformations.
Sensor ID (NITF)	I	10	0..2147483647	A unique identifier of a sensor within a GTDB texture transmittal.
Sensor Name (NITF)	S	20	--	Name of the sensor used to capture the image, e.g., "LANDSAT-2 RBV" or "MINOLTA 7S-II".
Sensor Type (NITF)	E	18	FRAME, MECHANICAL_SCANNER, PANORAMIC, PUSHBROOM, STRIP, OTHER	Type of sensor used to capture the image (e.g., any ordinary camera or a metric camera would be FRAME, SPOT satellite would be PUSHBROOM, and LANDSAT satellite would be MECHANICAL_SCANNER).
Sensor Types Supported	B, B, B	5	T, F; T, F; T, F	A set of three Boolean flags indicating whether a model or a FACS attribute can be used for specific sensor simulations. The flags are given in the order (radar, visual, infrared), and are separated by ASCII nulls.
Sensors Supported	B, B, B	5	T, F; T, F; T, F	Same as Sensor Types Supported.

TABLE A-III

Fieldname	Type	Length	Range	Description
Separation Plane Coefficients	R4D6	51	-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	The A, B, C, and D coefficients of the separating plane equations of the form Ax+By+Cz+D=0. Each coefficient is a 12-character Real number, separated from the other coefficients by an ASCII null.
Separation Plane Number	I	6	1..32767	The unique identifying number of a separation plane within a model.
Separation Planes Flag	B	1	T, F	A user-defined CDBTP parameter indicating whether separation planes are to be included with the models.
Shading Type	E	6	FIXED, FLAT, SMOOTH	Indicates type of shading to be used when rendering a polygon.
Shadow Minimization Flag (NITF)	B	5	TRUE, FALSE	Flag indicating whether operations to minimize the effects of shadows within an image have been performed.
Shape Code (Point Light String)	E	26	CYLINDER, RECTANGULAR_ PARALLELEPIPED, SPHERE_ HEMISPHERE, PYRAMID, CONE, OTHER	General 3-D shape of a feature.
Shape Code (Terrain Polygon Record)	E	8	TRIANGLE	Shape of terrain polygon.

TABLE A-III

Fieldname	Type	Length	Range	Description
SLOD Count	I	3	1..16	The number of Simulator Levels of Detail defined for the GTDB.
SLOD ID	I	3	0..16	The unique identifying number for a Simulator Level of Detail (SLOD) within a GTDB.
SMC/FDC Look Up Table Existence Flag	B	1	T, F	Parameter flag indicating the existence of a look up table for the values in SMC/FDC textures. When this flag is true, SMC/FDC textures file shall contain pointers into a look up table.
SMC/FDC Texture Existence Flag	B	1	T, F	The flag indicating the existence of SMC/FDC texture within an area block.
SMC/FDC Texture Resolution	R6	12	0.0..1.93428e+25	Approximate resolution of SMC/FDC texture in units of Meters/texel.
Source Agency/Project (NITF)	S	16	--	Name of the agency or project that created the digital source, e.g., "SOFATS", "P2851", etc.
Source Date (NITF)	S	6	YYMMDD	Date the digital source was created, where YY = Year, MM = Month, DD = Day.
Source ID (NITF)	I	10	0..2147483647	Unique identifier of an entry in the data source table.
Source ID Number	I	6	0..32767	The unique identifying number of a data source used to populate the SSDB.
Source Name (NITF)	S	20	--	Name of the original source, e.g., "EOSAT", "General Electric", etc.

TABLE A-III

Fieldname	Type	Length	Range	Description
Source Simulator	S	4	--	Identifies the particular simulator for which a model was created, if not generic.
Source Type (NITF)	E	1	H, S	Flag indicating hardcopy ("H") or softcopy ("S") source.
South Tile Neighbor ID (NITF)	I	10	0..2147483647	The identifier of the neighboring areal specific image to the south.
Special Environmental Conditions (NITF)	S	80	--	Textual description about any special conditions associated with an image.
Special Environmental Conditions Preference	S	80	--	Textual description about any preference for special conditions associated with an image.
Specific or Generic Texture Flag (NITF, non-NITF)	E	16	SPECIFIC_TEXTURE, GENERIC_TEXTURE	Flag indicating whether a texture is specific (derived from actual photo or satellite image) or generic (synthetic)
Specular	B	1	T, F	Flag indicating whether a feature or model polygon has the quality of being mirror-like. Synthetic default values are generated by P2851 software as a function of Feature Descriptor Code.
SSDB Culture LOD	E	5	LOD_0, LOD_1, LOD_2, LOD_3, LOD_4, LOD_5	The culture Level of Detail in the SSDB from which the GTDB culture is to be extracted.
Stage 1 Specific Textures Storage Size	R10	16	0.. 9.999999999E+99	The total number of bytes for all Stage 1 Specific Textures in a GTDB database.

TABLE A-III

Fieldname	Type	Length	Range	Description
Stage 2 Specific Textures Storage Size	R10	16	0.. 9.999999999E+99	The total number of bytes for all Stage 2 Specific Textures in a GTDB database.
Stage 3 Generic Textures Storage Size	R10	16	0.. 9.999999999E+99	The total number of bytes for all Stage 3 Generic Textures in a GTDB database.
Stage 3 SMC/FDC Textures Storage Size	R10	16	0.. 9.999999999E+99	The total number of bytes for all Stage 3 SMC/FDC Textures in a GTDB database.
Stage 3 Specific Textures Storage Size	R10	16	0.. 9.999999999E+99	The total number of bytes for all Stage 3 Specific Textures in a GTDB database.
Stage 4 Generic Textures Storage Size	R10	16	0.. 9.999999999E+99	The total number of bytes for all Stage 4 Generic Textures in a GTDB database.
Stage 4 SMC/FDC Textures Storage Size	R10	16	0.. 9.999999999E+99	The total number of bytes for all Stage 4 SMC/FDC Textures in a GTDB database.
Stage 4 Specific Textures Storage Size	R10	16	0.. 9.999999999E+99	The total number of bytes for all Stage 4 Specific Textures in a GTDB database.
Stage 5 Generic Textures Storage Size	R10	16	0.. 9.999999999E+99	The total number of bytes for all Stage 5 Generic Textures in a GTDB database.
Stage 5 SMC/FDC Textures Storage Size	R10	16	0.. 9.999999999E+99	The total number of bytes for all Stage 5 SMC/FDC Textures in a GTDB database.

TABLE A-III

Fieldname	Type	Length	Range	Description
Stage 5 Specific Textures Storage Size	R10	16	0.. 9.999999999E+99	The total number of bytes for all Stage 5 Specific Textures in a GTDB database.
Standard Image Filter Code n (NITF)	S	3	--	This field is reserved for future use.
Starting FDC Code	S	5	--	The Feature Descriptor Code used to define the beginning of a range of FDCs within various parameters of the CDBTP.
Storage Size	R10	16	0.. 9.999999999E+99	The total number of bytes for a single texture file.
Sun Azimuth (NITF)	R6	12	0.0..360.0	The clockwise angle measured in the horizontal plane, at the observer, between due north and the vertical projection of the center of the sun onto the horizon (expressed in degrees).
Sun Elevation (NITF)	R6	12	-90.0..90.0	The angle measured in a vertical plane, at the observer, between the horizon and the center of the sun, where negative values are below the horizon (expressed in degrees).
Superfeature Number	I	11	0..2147483647	The Feature Number of the original feature of which the current feature was a part prior to fragmentation.

TABLE A-III

Fieldname	Type	Length	Range	Description
Surface Material Category	I	4	1..14	Categorization of features based on the predominant material(s) making up the surface of the feature. Codes as defined by DMA in its DLMS Product Specification, Second Edition.
Surface Material Subtype	I	4	0..255	Codes available to further refine the Surface Material Category into subtypes.
Synthetic Culture Flag	B	1	T, F	User-defined CDBTP parameter indicating whether SSDB culture features marked as synthetic should be included in the GTDB.
Synthetic Data Flag (Features)	E	15	SYNTHETIC, REAL, EXPANDED_LINEAR	Flag indicating whether and what types of SSDB culture features marked as synthetic have been included within the given area block.
Synthetic Data Flag (FACS Attribute Record)	B	1	T, F	Flag indicating whether the FACS record consists of synthetic data.
Synthetic Data Flag (NITF)	B	5	TRUE, FALSE	Flag indicating whether the image source consists of synthetic data.
System Type (NITF)	S	6	--	Reserved for future use.
Tangent Point Height	R10	16	-9.999999999E+99.. 9.999999999E+99	A user-defined CDBTP parameter indicating the tangent point height to be applied during projection transformations.

TABLE A-III

Fieldname	Type	Length	Range	Description
Tape ID	S	10	--	A GTDB tape identifier used by P2851 software for archival configuration management. This ID does not refer to the tape on which a GTDB is distributed.
Tape Option	E	12	EVERYTHING, UPDATES_ONLY	Indication of whether the GTDB tape contains all area blocks of just those updated or added since the previous GTDB version.
Target ID (NITF)	S	17	BBBBBBBBBBFFFFFCC	The identification of the target, consisting of 10 characters of Basic Encyclopedia (BE), 5 characters of functional category code, and the 2 character country code as specified by FIPS-PUB 10-3.
Terrain Grid Area Block Flag	B	1	T, F	Boolean flag indicating whether a Terrain Grid Area Block (TGAB) pseudo-file exists within the given area block.
Terrain Grid Flag	B	1	T, F	User-defined CDBTP parameter indicating whether gridded terrain is desired in this GTDB.
Terrain Grid Source Flag	I	2	0..2	User-defined CDBTP parameter indicating whether the elevation values in the TGAB should be derived from the SSDB or from the terrain polygons in a TPAB being concurrently generated by the CDBTP. (0 = SSDB; 1 = GTDB 1; 2 = GTDB 2)

TABLE A-III

Fieldname	Type	Length	Range	Description
Terrain Grid Source SLOD	I	3	0..16	User-defined CDBTP parameter indicating, when the Terrain Grid Source Flag is set to > 0, from which SLOD the terrain polygons should be used to generate the terrain grid.
Terrain LOD	E	5	LOD_0, LOD_1, LOD_2, LOD_3, LOD_4	User-defined CDBTP parameter indicating which terrain LOD of the SSDB should be used as the terrain data source for a given area block.
Terrain Polygon Area Block Flag	B	1	T, F	Boolean flag indicating whether a Terrain Polygon Area Block (TPAB) pseudo-file exists within the given area block.
Terrain Polygon ID	I	11	1..2147483647	The unique identifying number for a terrain polygon within an area block.
Terrain Polygon Overlap Flag	B	1	T, F	A flag set to indicate that a model overlaps a terrain polygon boundary.
Terrain Polygons Flag	B	1	T, F	User-defined CDBTP parameter indicating whether polygonized terrain is desired in this GTDB.
Terrain Roughness Index	R6	12	0.0..1.93428E+25	A metric indicating the variability of the terrain within an area block.

TABLE A-III

Fieldname	Type	Length	Range	Description
Terrain Type	S	1	--	For generic terrain, a code indicating the general nature of the terrain (e.g., flat, rolling).
Texel Value (NITF)	BIN	--	--	The value in binary format of a given texel (texture element) within an NITF Image Data File; the number of bits in the value is determined by "Number of Bits Per Pixel".
Texture Data Format	S	80	--	Description of the texture data format used for a given texture in a GTDB. For most textures, the value "NITF" shall be used; for all Stage 1 textures not in NITF format, a description of the source format shall be provided here.
Texture Description (NITF)	S	80	--	Textual description of texture.
Texture Format	E	17	BAND_SEQUENTIAL, BAND_INTERLEAVED,	The ordering of the texture data by bands and pixels (texels).
Texture ID	I	11	0..2147483647	Unique ID number assigned to a texture within a GTDB texture library (same as GTDB Texture ID used by NITF).
Texture Map Reflectance	R6	12	0.0..1.0	Reflectance value assigned to a texture map.
Texture Mapping Set ID	I	11	0..2147483647	ID number identifying a set of textures used together when mapping (e.g., a summer texture set and a winter texture set may exist).

TABLE A-III

Fieldname	Type	Length	Range	Description
Texture Mapping Type	E	16	GLOBAL_BASED, MODEL_BASED, FACE_BASED, VERTEX_TO_VERTEX, NON_MAPPED	Method used in mapping texture onto terrain, culture, and models.
Texture Origin	I2D	13	0..99999, 0..99999	Location designated as the origin within a texture.
Texture Pattern Coordinates	I2D	13	0..99999, 0..99999	Positions within an image that are to be tied to the vertices of a model polygon when performing a vertex-to-vertex texture mapping.
Texture Reference Table Index	I	6	0..65535	A pointer to a texture reference in a texture reference table.
Texture Scale	R2D6	25	0.0..1.93428E+25, 0.0..1.93428E+25	Scale parameters applied to a texture map.
Texture Type (NITF)	E	14	RGB, INTENSITY, MULTI_SPECTRAL, SMC_FDC	The type of a texture.
Three-D Geometric Correction Flag (NITF)	B	5	TRUE, FALSE	Flag indicating whether a texture has been positioned/corrected (orthorectified) in 3D space.
Translation (Model Reference)	I2D		-324000000.. 324000000, -648000000.. 648000000	A set of coordinates defining vectors for local offset of a model from its nominal position.

TABLE A-III

Fieldname	Type	Length	Range	Description
Translation (Subsidiary Model Ref.)	R2D6		-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	A set of coordinates defining vectors for local offset of a subsidiary model from its nominal position.
	R3D6		-1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25, -1.93428E+25.. 1.93428E+25	
Translucency	R6	12	0.0..100.0	The degree to which a surface is transparent to visible light, expressed as a percentage. Synthetic default values are generated by P2851 software as a random function of Surface Material Category.
Transmissivity	R6	12	0.0..1.0	Ratio of radiant energy transmitted by a feature or model to the energy incident upon it. Synthetic default values are generated by P2851 software as a random function of Surface Material Category.
Two-D Geometric Correction Flag (NITF)	B	5	TRUE, FALSE	Flag indicating whether a texture has been positioned/corrected (geopositioned) in 2D space.

TABLE A-III

Fieldname	Type	Length	Range	Description
Type of Reference	E	7	FEATURE, MODEL	When culture has been fragmented onto terrain, indicates whether a particular culture reference record represents a feature or a model.
UL Corner X/Y Image Coordinates (NITF)	I2D	15	-999999.. 999999, -999999, 999999	X/Y cartesian coordinates of the upper left corner of the image.
UR Corner X/Y Image Coordinates (NITF)	I2D	15	-999999.. 999999, -999999, 999999	X/Y cartesian coordinates of the upper right corner of the image.
Use Models Flag	B	1	T, F	User-defined CDBTP parameter indicating whether model references should replace culture features, wherever an applicable model is available in the SSDB.
User Option	E	18	VIOLATE_FACE_COUNT, MEET_FACE_COUNT, ABORT_RUN	User-defined CDBTP parameter indicating the desired action if there is a conflict between user-specified parameters for culture selection and for maximum culture polygon count.

TABLE A-III

Fieldname	Type	Length	Range	Description
Value	I	6	0..32767	The value component of a color defined by the Hue-Chroma-Value model. P2851 normalizes Value to a range of 0 (black) to 32767 (white) instead of the familiar 0-100 percent lightness range. Synthetic default values are generated by P2851 software as a random function of Surface Material Category.
Vertex List Position	I	11	0..2147483647	An index into a coordinate list, identifying a coordinate used as a model, terrain, or culture vertex.
Vertex Normals Flag	B	1	T, F	User-defined CDBTP parameter indicating whether vertex normals should be calculated for terrain polygons.
Vertex-to-Vertex Mapping Flag	B	1	T, F	Parameter indicating the existence of vertex-to-vertex mapping parameters for texture.
Vertical Captured Texel Size (NITF)	R10	16	0.0.. 9.999999999E+99	Approximate ground distance for a texel (expressed in meters) in the vertical y-direction.
Vertical Resolution (NITF, non-NITF)	R6	12	0.0..1.93428e+25	Vertical length of a texel in meters, (e.g., 1.0 M/texel)
Vertical Size (NITF)	R6	12	0.0..1.93428e+25	The vertical size of the entire image, e.g., 1000.0 Meters.

TABLE A-III

Fieldname	Type	Length	Range	Description
Vertical Tile Block Size	I	11	0..2147483647	The number of texels in the vertical direction (column) in a texture block. A texture is divided into a whole number of blocks, each consisting of the same dimensions.
Visible Range	I	11	0..2147483647	Distance that a lignt feature can be seen, expressed in ten thousandths of arc seconds.
West Tile Neighbor ID (NITF)	I	10	0..2147483647	The identifier of the neighboring areal specific image to the west.
Width	R10	16	0.0.. 9.999999999E+99	The short dimension of a linear, point, point light, or point light string feature.
Wrap	B; B; B; B;	7	T, F; T, F; T, F; T, F	Flag indicating whether a texture pattern can be wrapped along its left, right, top and bottom edges while maintaining a "seamless" appearance.
Wrap Type	E	7	NO_WRAP, NORMAL, MIRROR	Flag indicating type of texture wrapping performed (for NORMAL, right edge aligned with left edge or top edge aligned with bottom edge; for MIRROR, each texture instance is mirrored from the previous texture instance).

This page intentionally left blank.

APPENDIX B

This page intentionally left blank.

FEATURE DESCRIPTOR CODES

10 **SCOPE.** This appendix identifies all Feature Descriptor Codes (FDCs) used within a Generic Transformed Data Base.

20. APPLICABLE DOCUMENTS

This section does not apply.

30. DEFINITIONS AND ACRONYMS

This section does not apply.

40. GENERAL REQUIREMENTS

40.1 **Feature Descriptor Codes.** Cultural features represented within the GTDB will be classified using a Feature Descriptor Code (FDC) that is an extension of the Defense Mapping Agency's Feature Attribute Coding Standard (FACS). The FACS uses a hierarchical system of five-character alphanumeric feature type codes. The first two characters are used to categorize a feature within a hierarchy of types. The last three characters are used to distinguish specific instances of the general type. The GTDB will use FACS codes as FDCs wherever an appropriate FACS code has been defined. In a few cases, it has been necessary to invent FDCs for feature types required by the GTDB but not presently defined by FACS.

40.1.1 FDC Categories Defined In DMA FACS. The following is a list of GTDB feature categories defined in FACS.

1	<u>Culture</u>
1A	Extraction
1B	Disposal
1C	Processing Industry
1D	Power Generation
1F	Associated Industrial Structures
1H	Institutional/Governmental
1I	Residential
1J	Agriculture
1K	Recreational
1L	Miscellaneous Features
1M	Storage
1N	Transportation R/R
1P	Transportation/Roads
1Q	Associated Transportation
1R	Air Traffic Services
1T	Communication/Transportation
1U	Airports
2	<u>Hydrography</u>
2A	Coastal Hydro
2B	Ports and Harbors
2C	Navaids
2D	Dangers and Underwater Features
2E	Depth Information
2F	Bottom Features
2G	Tide and Current Information
2H	Inland Water
2I	Miscellaneous Inland Water
2J	Snow/Ice
3	<u>Hypsography</u>
3A	Relief Portrayal

4	<u>Physiography</u>
4A	Exposed Surface Material
4B	Landforms
4C	Cross Country Movement
5	<u>Vegetation</u>
5A	Cropland
5B	Rangeland
5C	Woodland
5D	Wetlands
5E	Miscellaneous Vegetation
6	<u>Demarcation</u>
6A	Boundaries/Limits/Zones (Topographic)
6C	Boundaries/Limits/Zones (Hydrographic)
9	<u>General</u>
9B	Control Points
9C	Magnetic Variation Info
9D	Miscellaneous

40.1.2 **FDC Categories Defined By GTDB.** The following is a list of GTDB feature categories not defined in FACS. By convention, these categories use three-character designators rather than two. This was done to clearly differentiate GTDB-specific FDCs from FACS codes.

	<u>GTDB Specific Codes</u>
MOD	Model Specific Features
LGT	Lighting Features
ANT	Antenna Features
AIR	Miscellaneous Airport Features
MIS	Miscellaneous Features

40.1.3 **List of Supported FDCs.** The following is a list of Feature Descriptor Codes (FDCs) supported by GTDB. In most cases, the specific codes are taken directly from the DMA FACS Glossary. The GTDB includes FDCs for features which are not presently defined in FACS.

1	Culture
1A	Extraction
1A000	Extraction (General)
1A010	Mine
1A030	Quarry
1A031	Quarry Sheer-Wall
1A040	Rig/Superstructure
1A050	Well
1B	Disposal
1B000	Disposal Site/Waste Pile
1B010	Wrecking Yard/Scrap Yard
1C	Processing Industry
1C000	Processing Plant/Treatment Plant
1C010	Blast Furnace
1C020	Catalytic Cracker
1C030	Settling Basin/Sludge Pond
1D	Power Generation
1D010	Power Plant
1D020	Solar Panel
1D030	Substation/Transformer Yard
1F	Associated Industrial Structures
1F000	Associated Industrial Structures
1F010	Chimney/Smokestack
1F020	Conveyor
1F030	Cooling Tower
1F040	Crane
1F050	Dredge,Powershovel,Dragline
1F060	Engine Test Cell
1F070	Flare Pipe
1F080	Hopper

1H	Institutional/Governmental
1H020	Battery
1H050	Fort
1H075	Firing Range
1I	Residential
1I020	Mobile Home, Mobile Home Park
1J	Agriculture
1J030	Feed Lot
1J050	Windmill/Windmotor
1K	Recreational
1K000	Recreational
1K020	Amusement Park Attraction
1K030	Amusement Park
1K040	Athletic Field
1K060	Campground/Campsite
1K070	Drive-In Theater
1K080	Drive-In Theater Screen
1K090	Fairgrounds
1K100	Golf Course
1K110	Grand Stand
1K115	Outdoor Theater/Amphitheater
1K120	Park
1K130	Race Track
1K140	Recreational Vehicle Area
1K150	Ski Jump
1K160	Stadium/Amphitheater
1K170	Swimming Pool
1K180	Zoo

1L	<u>Miscellaneous Features</u>
1L005	Animal Sanctuary
1L015	Building
1L018	Building Superstructure Addition
1L020	Built-up Area
1L025	Cairn
1L030	Cemetery
1L040	Cliff Dwelling
1L050	Display Sign
1L060	Dragon (Tiger) Teeth
1L070	Fence
1L073	Flagstaff/Flagpole
1L075	Gallery
1L085	Geophysical Prospecting Grid
1L095	Homogeneous Radar Significant Area
1L100	Hut
1L105	Intelligence Annotation
1L110	Light Standard
1L115	Miscellaneous Vertical Obstruction
1L120	Missile Site
1L130	Monument
1L135	Native Settlement
1L140	Nuclear Accelerator
1L155	Overhead Obstruction Location
1L160	Pipeline/Pipe
1L165	Pipeline/Pipe Crossing Point
1L170	Plaza/City Square
1L180	Pumping Station
1L200	Ruins
1L208	Shantytown
1L210	Snow Shed/Rock Shed
1L215	Special Area
1L220	Steeple
1L228	Tent Dwellings
1L240	Tower (Non-Communication)
1L250	Underground Dwelling
1L260	Wall

1M	<u>Storage</u>
1M101	Depot (Storage)
1M020	Grain Bin
1M030	Grain Elevator
1M040	Mineral Pile
1M050	Silo
1M060	Storage Bunker/Storage Mound
1M070	Tank
1M080	Water Tower
1N	<u>Transportation R/R</u>
1N010	Railroad Track
1N050	R R Siding/R R Spur
1N075	R R Turntable
1N080	R R Yard
1N090	Tramway/Incline Railway
1P	<u>Transportation/Roads</u>
1P010	Cart Track
1P020	Interchange
1P030	Road
1P050	Trail

1Q	<u>Associated Transportation</u>
1Q010	Aerial Cableway Line/Ski Lift Line
1Q020	Aerial Cableway Pylon/Ski Lift Pylon
1Q040	Bridge/Cverpass/Viaduct
1Q045	Bridge Span
1Q050	Bridge Superstructure
1Q058	Constriction
1Q060	Control Tower
1Q065	Culvert
1Q068	Drop Gate/Rolling Block
1Q070	Ferry Crossing
1Q080	Ferry Site/Ferry Slip
1Q100	Distance Marker
1Q110	Mooring Mast
1Q111	Prepared Raft or Float Bridge Site
1Q115	Rest Area/Vehicle Stopping Area
1Q116	Route Marker
1Q118	Sharp Curve
1Q131	Tunnel
1Q132	Tunnel Entrance-Exit
1Q140	Vehicle Storage/Vehicle Parking
1R	<u>Air Traffic Services</u>
1R005	Air Obstruction Light
1R010	Airspace
1R030	Navaids (Aeronautical)
1R035	Radar Reflector
1R050	Route (Air)
1R060	Waypoint
1T	<u>Communication/Transportation</u>
1T005	Cable
1T010	Disk/Dish
1T020	Early Warning Radar Site
1T030	Power Transmission Line
1T040	Power Transmission Pylon
1T045	Radar Transmitter
1T050	Station (Communication)
1T060	Telephone Line/Telegraph Line
1T070	Telephone Pylon/Telegraph Pylon
1T080	Tower (Communication)

1U	Airports
1U025	Aircraft Landing Pad
1U030	Aircraft Facility
1U040	Aircraft Facility Beacon
1U045	Aircraft Facility Reference Point
1U050	Approach Lighting
1U060	Apron/Hardstand
1U070	Arresting Gear
1U080	Blast Barrier
1U120	Launch Pan
1U130	Overrun/Stopway
1U150	Revetment (Airfield)
1U160	Runway
1U190	Seaplane Landing or Take-Off Area
1U200	Taxiway
2	Hydrography
20000	Hydrography
2A	Coastal Hydro
2A010	Coastal Shoreline
2A020	Foreshore
2A040	Open Water (except inland)

2B	<u>Ports and Harbors</u>
2B010	Anchorage
2B020	Berth
2B030	Bollard
2B040	Breakwater
2B050	Calling-In Point
2B080	Dolphin
2B090	Dock
2B100	Fishery/Fish Stakes
2B105	Fishing Harbor
2B110	Fish Trap/Fish Weir
2B115	Gridiron
2B140	Jetty
2B150	Landing Place
2B155	Maritime Station
2B160	Mooring Ring
2B170	Offshore Loading Facility
2B180	Oyster or Cultivated Shellfish Bed
2B190	Pier, Wharf
2B220	Ramp
2B230	Sea Wall
2B240	Slipway

2C	<u>Navaids</u>
2C000	Navaids
2C010	Buoy
2C020	Clearing Line
2C030	Electronic Beacon
2C040	Leading Line
2C050	Light
2C055	Marker
2C060	Visual Beacon

2D	<u>Dangers and Underwater Features</u>
2D000	Miscellaneous Underwater Features
2D010	Breakers
2D020	Crib
2D030	Discolored Water
2D040	Eddies
2D050	Foul Ground
2D060	Kelp
2D080	Overfalls/Tide Rips
2D090	Perches/Stakes
2D100	Piling
2D110	Platform
2D120	Reef
2D125	Reef Pool
2D130	Rock
1D140	Snag/Stump
2D180	Wreck
2E	<u>Depth Information</u>
2E010	Depth Curve
2E015	Depth Contour
2E020	Sounding
2E030	Track Line
2E040	Track Swath
2F	<u>Bottom Features</u>
2F010	Bottom Characteristics
2G	<u>Tide and Current Information</u>
2G010	Current Arrow/Flow Arrow
2G020	Tide Gauge
2G030	Tide Data Point
2G040	Current Diagram

2H	<u>Inland Water</u>
2H000	Inland Water
2H010	Aqueduct
2H020	Canal
2H030	Ditch
2H040	Filtration/Aeration Beds
2H050	Fish Hatchery
2H060	Flume
2H070	Ford
2H075	Inland Shorline
2H080	Lake/Pond
2H090	Land Subject to Inundation
2H095	Miscellaneous Surface Drainage Feature
2H100	Moat
2H110	Penstock
2H120	Rapids
2H130	Reservoir
2H140	River/Stream
2H145	River or Stream Vanishing Point
2H150	Salt Evaporator
2H160	Sebkha
2H170	Spring/Water-Hole
2H180	Waterfall
2I	<u>Miscellaneous Inland Water</u>
2I010	Cistern
2I020	Dam
2I030	Lock
2I040	Sluice Gate
2I050	Water Intake Tower
2J	<u>Snow/Ice</u>
2J000	Snow/Ice
2J020	Glacial Moraine
2J030	Glacier
2J040	Ice Cliff
2J060	Ice Peak, Nunatak
2J065	Ice Shelf
2J070	Pack Ice
2J080	Polar Ice
2J100	Snow Field/Ice Field
2J110	Tundra

3	<u>Hypsography</u>
3A	<u>Relief Portrayal</u>
3A010	Contour (Land)
3A020	Ridge Line
3A030	Spot Elevator
4	<u>Physiography</u>
4A	<u>Exposed Surface Material</u>
4A005	Asphalt Lake
4A010	Ground Surface
4A015	Cleared Way
4A020	Salt Pan
4A030	Surficial Material
4B	<u>Landforms</u>
4B000	Landforms
4B010	Bluff/Cliff, Escarpment
4B030	Cave
4B060	Crevice/Crevasse
4B070	Cut
4B071	Cut Line
4B080	Depression
4B090	Embankment
4B100	Esker
4B110	Fault
4B115	Geothermal Feature
4B135	Island
4B145	Miscellaneous Obstacle
4B150	Mountain Pass
4B155	Ravine/Gorge/Canyon
4B160	Rock Strata, Rock Formation
4B170	Sand Dune/Sand Hill
4B176	Slope Category
4B180	Volcano
4B190	Volcano Dike
4C	<u>Cross Country Movement</u>
4C010	Cross Country Movement Speed Range

5	<u>Vegetation</u>
50000	Vegetation
5A	<u>Cropland</u>
5A010	Cropland (Cultivated)
5A020	Hedgerow
5A030	Nursery
5A040	Orchard/Plantation
5A050	Vineyard/Hops
5B	<u>Rangeland</u>
5B010	Grassland
5B020	Scrub/Brush
5C	<u>Woodland</u>
5C010	Bamboo/Cane
5C015	Firebreak
5C020	Oasis
5C030	Trees
5D	<u>Wetlands</u>
5D010	Bog
5D020	Hummock
5D030	Swamp
5D040	Marsh
5E	<u>Miscellaneous Vegetation</u>
5E010	Land Use/Land Cover (Vegetation)

6	Demarcation
6A	Boundaries/Limits/Zones (Topographic)
6A000	Administrative Boundary
6A010	(Omitted)
6A020	Armistice Line
6A030	Cease-Fire line
6A040	Claim Line
6A050	Convention Line/Mandate Line
6A060	Defacto Boundary
6A070	Demilitarized Zone
6A090	(Omitted)
6A110	International Date Line
6A140	(Omitted)
6A170	Zone of Occupation
6C	Boundaries/Limits/Zones (Hydrographic)
6C020	Colregs Demarcation Line
6C030	Customs Boundary
6C040	Dredged Channel/Dredged Area
6C070	Harbor Limit
6C075	Inshore Traffic Zone
6C090	Maritime Limit
6C100	Measured Distance Line
6C110	Mine Danger Area (Hydro Limit)
6C120	Prohibited Area (Hydro Only)
6C130	Radar Reference Line
6C150	Restricted Area (Hydro Only)
6C160	Roundabout
6C165	Route
6C170	Safety Fairway
6C175	(Omitted)
6C177	Swept Area
6C180	Traffic Separation Scheme
6C210	Work in Progress Area

9	<u>General</u>
9B	<u>Control Points</u>
9B030	Boundary Monument
9B035	Control Point
9B040	Diagnostic Point
9B045	Elevation Point
9B070	Precise Radar Significant Location
9C	<u>Magnetic Variation Info</u>
9C040	Magnetic Disturbance Area
9D	<u>Miscellaneous</u>
9D005	Area of RSPS Accuracy Limitations
9D015	Point of Change
9D020	Void Collection Area
9D022	Homogeneous Aggregate Feature
9D020	Dissimilar Aggregate Feature

	Model Specific Features
MOD01	Aircraft, (Static, On Ground)
MOD02	Armored Personnel Carriers (Static)
MOD03	Artillery
MOD04	Bomber Aircraft (Enemy, Static)
MOD05	Bomber Aircraft (Friendly, Static)
MOD06	Fighter Aircraft (Enemy, Static)
MOD07	Fighter Aircraft (Friendly, Static)
MOD08	Ground Vehicles (Enemy, Static)
MOD09	Ground Vehicles (Friendly, Static)
MOD10	Gun
MOD11	Helicopter (Enemy, Static)
MOD12	Helicopter (Friendly, Static)
MOD13	Missile (Static)
MOD14	Missile Transporter
MOD15	Mobile Anti-Aircraft Artillery
MOD16	Mobile Rocket Launcher (SAM Launcher)
MOD17	Radar Antenna, Vehicle Mounted (Static)
MOD18	Single Tree
MOD19	Surface Ship (Enemy)
MOD20	Surface Ship (Friendly)
MOD21	Tank (Vehicle, Static)
MOD22	Tents
MOD23	Transport/Tanker Aircraft (Enemy, Static)
MOD24	Transport/Tanker Aircraft (Friendly, Static)
MOD25	Truck
MOD26	Weapon Delivery Systems (General)

LGT	<u>Lighting Features</u>
LGT01	Airport Boundary Lighting
LGT02	Approach Light Framework
LGT03	Athletic Field Light
LGT04	Boundary Light
LGT05	Fishing Light
LGT06	Flood Lights
LGT07	Leading Light, Lights in Line
LGT08	Lighted Wind Indicator
LGT09	Obstruction Light
LGT10	Private Light
LGT11	Runway Centerline Light
LGT12	Runway Overrun Lighting
LGT13	Street Lamp
LGT14	Taxiway Light
LGT15	Terminal Light
LGT16	Threshold Light
LGT17	Touchdown Zone Lighting
ANT	<u>Antenna Features</u>
ANT01	Airport Surveillance Radar
ANT02	American Forces Radio Station Antenna
ANT03	Commercial Broadcasting Antenna
ANT04	Direction Finder
ANT05	Flight Service Station (FSS) Antenna
ANT06	Ground Control Intercept Antenna
ANT07	Radar Antenna with Radome
ANT08	Radar Antenna, Tower Mounted with Radome
ANT09	Radio Relay Mast
ANT10	Remote Communications Outlet (RCO) Antenna
ANT11	Stick Mast
ANT12	Telemetry Antenna
ANT13	U.S. Weather Station Antenna
AIR	<u>Miscellaneous Airport Features</u>
AIR01	Fuel Handling Facility
AIR02	Refueling Units
AIR03	Runway Distance Remaining Marker
AIR04	Runway Markings
AIR05	U.S. Weather Station
AIR06	Wind Indicator

MIS	<u>Miscellaneous Features</u>
MIS01	Cinder Cones
MIS02	Farm Cluster
MIS03	Oil Storage Pit
MIS04	Outfalls
MIS05	Pump Out Facility
MIS06	Regional Feature
MIS07	Sludge Gate
MIS08	Watering Place
UNKWN	Unknown

This page intentionally left blank.

APPENDIX C

This page intentionally left blank.

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

FIGURES

RATIONALE AND GUIDANCE

(This appendix is not a mandatory part of this standard. The information contained herein is intended for guidance only.)

10 SCOPE

10.1 Scope/Background

10.1.1 The Generic Transformed Data Base (GTDB) is intended to serve as a data base which will support real-time visual/infrared and radar sensor simulator systems. The single GTDB format documented in this data base design document is a superset of data base features contained in the visual/infrared data bases, as well as radar data bases. This single format is expected to reduce software development and maintenance costs among users of the GTDB.

10.1.2 It should be emphasized that, even though there is a single format for a GTDB, there can be great variation in content from GTDB to GTDB. A GTDB generated for a radar simulator may continue to be significantly different from a GTDB for a visual system in terms of feature content, levels of detail, and terrain representation. However, one long-term advantage of a common format is that, as image generator technology advances make it feasible, the GTDB will be able to support integrated multi-sensor simulations from a common data base.

10.1.3 The GTDB will be generated as needed from current terrain, culture, model, and photo texture data maintained within the DoD Standard Simulator Data Base (SSDB). The software subsystem which transforms standard simulator data base data into a GTDB is designated the Common Data Base Transformation Program (CDBTP). A highly parameterized common data base transformation program has been implemented which will permit tailoring of GTDBs to the special requirements of a particular simulator or training mission, as well as the sensor and platform being simulated.

10.1.4 This tailoring of GTDBs is intended to minimize the complexity of user-written Formatter Programs (FPs) which will filter and format a GTDB for execution on a particular simulator system.

10.1.5 In addition to real data, the GTDB supports the extensive use of default values and synthetically generated data. The philosophy here is to have the GTDB supply common synthetic data to maintain correlation among training systems, rather than to deliver blank fields and have each training system independently assign its own defaults. Default values associated with particular fields are documented in Appendix A of this standard.

10.2 **Purpose.** Air Force Project 2851, also known as Standard DoD Simulator Digital Data Base/Common Transformation Program, is a Tri-Service program to develop standardized data bases and transformation software to be used by all future DoD training simulator applications. Currently, each simulator subsystem purchased by the DoD is delivered with its own data base and transformation program. This not only costs the Government in terms of recurring development costs associated with the "re-invention" of similar, if not identical products, but also results in proliferation of non-standard, incompatible data bases and software, all of which must be maintained by DoD. The Project 2851 effort is intended to improve this situation by developing common standards to be utilized throughout DoD.

10.2.1 Another major problem being addressed by Project 2851 is the lack of correlation between different sensor simulators (radar, infrared, out-the-window visuals) within a training system, as well as among training systems in a network. The lack of correlation can result in poor training or limit the scope of training programs. By developing standard data bases and software, Project 2851 is intended to improve simulator data base correlatability.

20 APPLICABLE DOCUMENTS

This section does not apply.

30 DEFINITIONS AND ACRONYMS

30.1 **Definitions.** For the purpose of this appendix, the following definitions apply.

30.1.1 Accuracy: degree of conformity of values in a data base to an established standard. Accuracy relates to the quality of the result of a measurement, and is distinguished from precision which relates to the quality of the operation by which the result was obtained.

30.1.2 Anomaly: (1) an entry in a data base that deviates from the expected norm. (2) In geodesy, a deviation of an observed value from a theoretical value due to a corresponding irregularity in the earth's structure at the area of observation.

30.1.3 Area Block: the fundamental physical unit of a gaming area data base, used to subdivide a large gaming area into manageable quantities of data. The size of an area block may vary from data base to data base and even between Simulator Levels of Detail (SLODs) within a data base.

30.1.4 Areal Feature: a cultural object represented within a data base as a polygon.

30.1.5 Areal Photo Texture: a two-dimensional array of data, typically from a digitized photograph, used to modulate the appearance of an areal feature across its surface.

30.1.6 Aspect Change: variations in the appearance of an object viewed by radar from varying directions due to changes in the effective reflecting area of the object.

30.1.7 Associated Feature: a feature or group of features which correspond(s) to another feature at a lower level of detail (LOD).

30.1.8 Base Map: a map or chart of known or assumed accuracy used as a reference for positioning additional data.

30.1.9 Cartesian Coordinates: a coordinate system in which locations of points in space are expressed by reference to three mutually perpendicular planes.

30.1.10 Cartographic: refers to the art and science of expressing graphically, by maps and charts, the known physical features of the earth.

30.1.11 Cartographic Displacement: the horizontal shifting of the plotted position of a map feature from its true position, caused by required adherence to prescribed line weights and symbol sizes to promote readability.

30.1.12 Cell: the basic unit of geographic data coverage for terrain and culture within the SSDB. By convention, a typical cell represents a 1x1 degree area bounded by whole degrees of latitude and longitude.

30.1.13 Centerline: the continuous center of a linear feature such as a highway.

30.1.14 Central Meridian: the line of longitude at the center of a projection.

30.1.15 Chart: a special-purpose map, generally designed for navigation or other particular purposes, in which essential map information is combined with various other data critical to the intended use.

30.1.16 Clipping: the process of cutting and closing off line and areal features along some defined boundary. With areal features, the clipping process introduces an artificial line segment in order to close the feature and maintain structural integrity.

30.1.17 Collision Test Point: a special vertex in a 3-D dynamic (moving) model used to detect a collision of that model with terrain or other models situated within a simulator scene.

30.1.18 Common Data Base Transformation Program (CDBTP): the software subsystem within the Project 2851 System that processes data from the Standard Simulator Data Base (SSDB) and converts it into a format (GTDB) that is usable by simulators with a minimal amount of additional processing.

30.1.19 Configuration Management Data Base (CMDB): a collection of Project 2851 data base status and configuration information maintained by the Project 2851 Configuration Management Program (CMP).

30.1.20 Configuration Management Program (CMP): the software subsystem within the Project 2851 System that will track the configuration and status of the SSDB and the GTDBs.

30.1.21 Convex Feature: an areal feature all of whose vertices form exterior angles equal to or greater than 180 degrees.

30.1.22 Coordinate: an ordered set of real numbers specifying a point in two-dimensional (2-D) or three-dimensional (3-D) space.

30.1.23 Coordinate Conversion: the process of changing coordinate values from one representation (e.g., geographic lat/long) to another (e.g., UTM).

30.1.24 Coordinate System: a consistent scheme for designating relative locations of a set of points within a reference space.

30.1.25 Coordinate Transformation: a mathematical or graphic process of obtaining a modified set of coordinates, as in a map projection, by some combination of rotation of coordinate axes at their point of origin, translocation of the point of origin, modification of scale along coordinate axes, or change of the size or geometry of the reference space.

30.1.26 Correlated Data Bases: multiple simulator data bases which, by their structure and content, support an acceptable degree of correlation among multiple simulator displays.

30.1.27 Correlation: the correspondence and synchronization of multiple simulator sensor displays (e.g., visual and radar) over the same gaming area. At a minimum, correlation implies the absence of conflicting information across multiple displays.

30.1.28 Cue: any sensory perception which provides information that can be of use in maneuvering or operating an aircraft, vehicle, or weapon system.

30.1.29 Culture: strictly speaking, objects on the terrain that have been constructed by man; in practice, the term is often used synonymously with "feature," indicating natural as well as man-made objects.

30.1.30 Culture Decomposition: the process of breaking up a nonconvex areal feature into multiple convex features.

30.1.31 Culture Fragmentation: the process of breaking up a line or areal feature at the boundaries of underlying terrain polygons.

30.1.32 Data Base Generation/Modification Program (DBGMP): the software subsystem within the Project 2851 System that creates and maintains the Standard Simulator Data Base (SSDB), using digital and nondigital input products, interactive editing, and algorithmic processes.

30.1.33 Datum: in general, any numerical or geometric quantity or set of such quantities which are designated as a reference or base for other quantities. Within data bases of earth features such as the Project 2851 SSDB and GTDBs, horizontal and vertical datums are needed to position coordinates in 3-D space.

30.1.34 Digital Feature Analysis Data (DFAD): a standard data base product of the Defense Mapping Agency containing generalized, but rigidly specified, description and portrayal of cultural or planimetric data, expressed at different levels of feature density. Level 1 is roughly equivalent to the detail that could be derived from a map with a scale of one inch equals four miles (1: 250,000). Level 2 is roughly equivalent to a map scale of 1:50,000. Higher-resolution prototype and special versions of DFAD exist for very limited areas of coverage.

30.1.35 Digital Landmass System (DLMS): a standard DMA-produced data base containing feature (DFAD) and terrain elevation (DTED) data representing the landmass of the earth.

30.1.36 Digital Radar Landmass Simulator (DRLMS): a device which uses digital data bases to create simulations of real-beam radar returns.

30.1.37 Digital Terrain Elevation Data (DTED): a standard data base of terrain elevations produced by the Defense Mapping Agency. Level 1 DTED contains discrete elevations expressed at intervals of three arc-seconds of latitude (roughly 100 meters). Level 2 DTED is not regularly produced by DMA but is specified at intervals of one arc-second of latitude (roughly 30 meters).

30.1.38 DMA Feature File (DMAFF): a feature data coding standard drafted by DMA which has been superseded by FACS.

30.1.39 Drainage: all features associated with water, such as shorelines, rivers, lakes, marshes, etc.

30.1.40 Edge: a straight line segment connecting two vertices.

30.1.41 Elevation: a vertical distance from a datum, e.g., mean sea level.

30.1.42 Face: a polygonal surface represented within a graphic data base.

30.1.43 Feathering: the process of smoothing out the transition between adjoining or overlapping patches of feature data of differing densities.

30.1.44 Feature: a natural or man-made object of sufficient importance to express as an entity in a simulator data base.

30.1.45 Feature Attribute Coding Standard (FACS): a highly structured coding system developed by the Defense Mapping Agency for assigning alpha-numeric codes to describe a feature and its attributes.

30.1.46 Feature Classes: generic groupings of cultural and natural objects which can be further classified into sub-features providing more complete descriptors.

30.1.47 Filtering: the selective elimination of some data from a set of data.

30.1.48 Formatter Program (FP): a simulator-specific transformation program to convert a Project 2851 GTDB into the on-line format unique to the hardware.

30.1.49 Gaming Area: the geographic area of coverage of a GTDB or simulator on-line data base, representing the geographic boundaries of training simulations which can be supported with the data base.

30.1.50 Generic Radar Data Base (GRDB): a GTDB created to support radar simulation.

30.1.51 Generic Transformed Data Base (GTDB): (1) a sensor-specific gaming area data base generated by the CDBTP by transforming selected SSDB data into a form more readily usable by a simulator system. (2) Also, the software subsystem of the Project 2851 System that provides GTDB data handling functions.

30.1.52 Generic Visual/Infrared Data Base (GVIDB): a GTDB created to support visual and/or infrared simulation.

30.1.53 Geocentric Coordinates: a coordinate system which defines the position of points relative to the center of the earth.

30.1.54 Geodetic: refers to the science which deals with the determination of the size and figure of the earth.

30.1.55 Geodetic Coordinates: the quantities of latitude, longitude, and height, used to define the position of a point on the surface of the earth with respect to the reference spheroid.

30.1.56 Geographic Coordinates: the quantities of latitude and longitude used to define the position of a point on the surface of the earth with respect to the reference spheroid.

30.1.57 Goodness of Fit: a parameter or measure of degree of correspondence between polygonized terrain and the original elevation matrix values.

30.1.58 Gray Scale: a scale of tones ranging from white to black.

30.1.59 Ground Resolution: the minimum distance which can be detected between two adjacent features, or the minimum size of a feature expressed in size of objects or distances on the ground.

30.1.60 Homogeneous Area: an area of uniform radar reflectance.

30.1.61 Horizontal Datum: a set of geodetic parameters used to specify the shape of a reference ellipsoid to which positional coordinates are referenced. Due to the irregular shape of the earth (as well as to political and historical factors), different datums are commonly used in different parts of the world. For a worldwide data base such as the SSDB, a unified world datum (WGS84) will be used.

30.1.62 Image Generator (IG): a device for producing computer-generated scenes, capable of simulating real-time movement through the scene.

30.1.63 Instantaneous Field of View (IFOV): the portion of a large visual scene (e.g., as in a dome display) where an observer is actually looking at a moment in time. Some simulators use head or eye tracking devices to concentrate scene density within the IFOV.

30.1.64 Island: an area within a GTDB defined via transformation parameters as having data content or density characteristics different from the surrounding area; islands may be defined within other islands but may not overlap.

30.1.65 Lambert Conformal Conic: a map projection commonly used in aeronautical charting, on which geographic meridians are represented by straight lines which meet at a common point outside the limits of the map, and geographic parallels are represented by a series of arcs of circles having this common point for a center.

30.1.66 Level of Detail (LOD): a discrete level within a range of levels of SSDB content density, used to store alternate representations of any given area of coverage.

30.1.67 Linear Feature: a cultural object represented within a data base as one or more connected line segments.

30.1.68 Manuscript: a physical subdivision of an SSDB cell. Manuscripts cannot overlap each other.

30.1.69 Microdescriptors: a special class of multi-attribute data structures developed by the DMA to encode complex attributes in its high-resolution (e.g., Level V DFAD) digital cartographic data bases.

30.1.70 Model: a graphical or mathematical representation of an object.

30.1.71 Model Instancing: the placement of multiple references to a common model throughout a data base, as opposed to replicating the model at every location.

30.1.72 Model Photo Texture: a two-dimensional array of data, typically from a digitized photograph, used to modulate the appearance of a model surface.

30.1.73 Occultation: the hiding from view of one feature or surface by the interposition of another feature or surface along the line of sight.

30.1.74 Origin: the reference point relative to which all coordinates within a coordinate set are defined.

30.1.75 Paneling: the process of reconciling data along the boundary between feature or terrain data sets; also referred to as "edge matching" or "border matching."

30.1.76 Photo Texture: a technique for modifying the characteristics of a graphic feature surface based on information typically stored in a two-dimensional array of data, such as from a digitized photograph. The application of photo texture to areal culture or model polygons permits greatly increased feature detail and realism without additional complexity in the culture or model geometry.

30.1.77 Pixel: a contraction of the term, "picture element"; an element in a gridded representation of 2-D space.

30.1.78 Point Feature: a cultural object whose location within a data base is represented as a single point, or as a set of points.

30.1.79 Point Light Feature: a light-emitting object represented within a data base as a single point.

30.1.80 Point Light String Feature: a light-emitting object represented within a data base as a set of points.

30.1.81 Quadrangle: a rectangular, or nearly so, area covered by a map, usually bounded by given meridians of longitude and parallels of latitude.

30.1.82 Radar Correlation: the process of electronically relating real-time radar images with stored digital data on the position and radar reflectance of terrain and features, used for navigation and guidance.

30.1.83 Radar Target: an object which reflects a sufficient amount of a radar signal to produce an echo signal on the radar screen.

30.1.84 Real Time: in computer systems, the capability to detect, display, and/or react to an event as it actually occurs.

30.1.85 Real-World Data: data, such as DMA DFAD and DTED, derived from imagery or other sources known to represent actual real-world conditions.

30.1.86 Resolution: a measure of the ability to distinguish detail, as applied to the power of a sensing system, the sharpness of an image, or the level of detail in a data base.

30.1.87 Ridge Line: a graphic representation of a long narrow land elevation, typically used to represent terrain forms significant for low altitude radar prediction.

30.1.88 Roll In: the process of transferring a data file from off-line storage media to on-line storage.

30.1.89 Roll Out: the process of transferring a data file from on-line storage media to off-line storage.

30.1.90 Separation Plane: a geometric entity used to simplify dynamic resolution of visual rendering priorities of polygons within a 3-D model.

30.1.91 Simulator Level of Detail (SLOD): a discrete level within a range of levels of GTDB content density used to store alternate views of any given area of coverage and representing the effects of changes in proximity, sensor range, gain, or aspect angle.

30.1.92 Standard Linear Format (SLF): a feature data format developed by DMA for intermediate product exchange among its various production systems. It employs a link-node structure to describe feature planimetry, in which shared segments are digitized only once and referenced by multiple features, thus avoiding the slivers-and-gaps problems of the older polygonal DFAD format.

30.1.93 Standard Simulator Data Base (SSDB): (1) the central repository of validated standard digital feature, elevation, model, and photo texture data within the Project 2851 System. (2) Also, the software subsystem of the Project 2851 System that provides SSDB data handling functions.

30.1.94 Synthetic Breakup: the automatic or semi-automatic decomposition of large areal features into more detailed component point, line, areal, point light, and/or point light string synthetic features to support more realistic-looking simulation.

30.1.95 Synthetic Data: data derived algorithmically (e.g., by interpolation or synthetic enhancement), by manual enhancement, or from other sources without regard or reference to actual real-world conditions.

30.1.96 Synthetic Enhancement: the automatic or semi-automatic augmentation of existing low-resolution data with synthetic higher-resolution data to support more realistic-looking simulation, in the absence of real-world high-resolution data.

30.1.97 Temporal Effects: changes to cultural characteristics based on time of day, season, weather, or other conditional situations.

30.1.98 Terrain: the spatial configuration of the surface of the earth in 3-D space, typically represented in simulator data bases as a grid of terrain elevation posts or as a network of terrain polygons.

30.1.99 Terrain Leveling: the process of forcing terrain data to be "flat" (i.e., of uniform elevation) wherever the terrain corresponds to surface features which logically should be portrayed as level surfaces (e.g., lakes).

30.1.100 Terrain Polygon: a polygon situated in 3-D space, used to represent the slope of an area of the earth's surface.

30.1.101 Terrain Post: a single data element within a grid of terrain elevation values.

30.1.102 Tessellation: the process of forming a mosaic pattern using fundamental units or patterns of data.

30.1.103 Texture Mapping: the process of applying digital patterns or photographic images upon a graphic face to create a more realistic or detailed rendering of an object.

30.1.104 Thinning: the systematic reduction of the number of points defining a line or areal feature for purposes of simplification.

30.1.105 Topography: the configuration of the surface of the earth, including its relief (hypsography), drainage (hydrography), man-made features (culture), and vegetation.

30.1.106 Transformation Parameter: any of many control data values entered into the CDBTP at time of execution to tailor the resulting GTDB(s). Parameters may represent basic identification and control information, selections from a set of options, input variables used by numerical algorithms, or indications of relative priorities to control trade-off algorithms.

30.1.107 Transition Band: a limited area of data base coverage in which the data density is gradually changed to smooth the transition from high-detail data on one side of the band to low-detail data on the other.

30.1.108 Universal Transverse Mercator (UTM): a military grid coordinate system based on the transverse Mercator projection.

30.1.109 Utility Program (UP): the software subsystem within the Project 2851 System that provides utility services and common functions for the rest of the system.

30.1.110 Vector Format: a format whereby features in a data base are expressed as a series of connected lines or curves; the counterpart of "raster" format wherein discrete portions of the feature are expressed as data along a scan line.

30.1.111 Vertex: an ordered triplet (x,y,z) of real numbers specifying a point in 3-D space.

30.1.112 Vertical Datum: a level surface used as a reference from which to reckon elevation values. The Project 2851 data bases use Mean Sea Level.

30.1.113 World Geodetic System (WGS): a unified world geodetic datum based on a combination of all available astrogeodetic, gravimetric, and satellite tracking observations; especially significant in calibration and operations of an inertial guidance system for air-breathing and orbital platforms. The datum is periodically updated as warranted by new data, with the current datum designated as WGS84.

30.1.114 2-D Static Model: a planar graphic representation of a cultural object, used by simulators to render point, line, or areal surface features.

30.1.115 <u>3-D Dynamic Model</u>: a three-dimensional graphic representation of a cultural object, used by simulators to render point, line, or areal features having height and capable of motion within the simulated scene.

30.1.116 <u>3-D Static Model</u>: a three-dimensional graphic representation of a cultural object, used by simulators to render point, line, or areal features having height.

3.2 **Acronyms**. For the purpose of this appendix, the following acronyms apply.

ANSI	American National Standards Institute
ASCII	American Standard Code Information Interchange
CDBTP	Common Data Base Transformation Program
CIG	Computer Image Generator
CM	Configuration Management
CMP	Configuration Management Program
COTS	Commercial Off-the-Shelf
CSCI	Computer Software Configuration Item
DB	Data Base
DBDD	Data Base Design Document
DBGMP	Data Base Generation/Modification Program
DBMS	Data Base Management System
DFAD	Digital Feature Analysis Data
DLMS	Digital Landmass System
DMA	Defense Mapping Agency
DRLMS	Digital Radar Landmass Simulator
DTED	Digital Terrain Elevation Data
EO	Electro-Optical
EOF	End of File
FAC	Feature Analysis Code
	Feature Attribute Code
FACS	Feature Attribute Coding Standard
FDC	Feature Descriptor Code
FID	Feature Identification Descriptor
FLIR	Forward-Looking Infrared
FP	Formatter Program
FPI	Frames per Inch
GB	Gigabyte(s)
GTDB	Generic Transformed Data Base
HVC	Hue-Value-Chroma
IG	Image Generator
I/O	Input/Output
IR	Infrared
KB	Kilobyte(s)
LOD	Level of Detail
LVT	Local Vertical Tangent
MB	Megabyte(s)
MBR	Minimum Bounding Rectangle
MC&G	Mapping, Charting, and Geodesy
MSL	Mean Sea Level
NOE	Nap of the Earth
QA	Quality Assurance
QC	Quality Control
RGB	Red-Green-Blue
SAR	Synthetic Aperture Radar
SDDD	Software Detailed Design Document

SLOD	Simulator Level of Detail
SMC	Surface Material Category
SNM	Square Nautical Miles
SSDB	Standard Simulator Data Base
S/W	Software
UP	Utility Program
USGS	United States Geological Survey
UTM	Universal Transverse Mercator
VMS	Virtual Memory System
WGS	World Geodetic System
W/S	Workstation

40 GENERAL REQUIREMENTS

40.1 **Media.** At the time of the initial publication of this standard (1992), all GTDBs are being distributed exclusively on nine-track tape. It is envisioned that, in the future, alternative media will be supported, such as eight millimeter tape or electro-optical disk; the DoD Simulator Data Base Facility will keep the user community apprised of such changes as they occur. In either case, nine-track tape will continue to be supported as long as it remains an industry-standard medium.

40.1.1 **Structure/Labeling.** Data units called "files" will be delineated as ANSI-standard tape files, as implemented under VAX/VMS. Data "records" within a file will be delineated as ANSI-standard variable-length tape records, blocked at 2048 bytes. Records will not span blocks.

40.1.2 **Multi-Volume Sets.** Pseudo-files, as well as physical files, will be allowed to span volumes.

40.2 **Content.** As a format-only standard, GTDB requires only a minimum amount of mandatory information, The content of a specific GTDB is not dependent on requirements defined in the standard, but rather on two other factors: (a) that information which is available in the data set from which the GTDB is derived, i.e., the SSDB; and (b) that information which is requested by the recipient of the GTDB.

40.2.1 **Field Population.** One of the few content restrictions imposed by the GTDB standard is that all eligible fields be populated whenever their parent records are included in a data set. This corresponds to factor (a) of paragraph 40.2, above.

40.2.1.1 **Valid Data.** Whenever a GTDB contains a particular field (based upon the above criteria), and the SSDB contains the information necessary to populate that field, the value of the field will contain that SSDB value (or a corresponding GTDB value). This rule will ensure that pertinent information is always made available to the application.

40.2.1.2 **Default Data.** In cases in which a GTDB contains a field which does not have a corresponding SSDB value, the field will be assigned a predetermined default value.

40.2.2 **Content Specifications.** Corresponding to factor (b) of paragraph 40.2, above, the content of a GTDB is dependent on the user's specifications. These specifications take the form of a number of transformation parameters, which are used to extract and manipulate SSDB information to generate a GTDB tailored to the intended application. These parameters are described in the following subparagraphs.

40.2.2.1 **Datum.** This identifies the horizontal reference for geodetically-based GTDBs.

40.2.2.2 **Coordinate System.** The user may select geodetic, geocentric, or projected coordinates.

40.2.2.3 **Terrain Representation.** There are three choices of terrain representation available in the GTDB. One of these is just a simple extraction of the grid-post values stored in the SSDB. The second is a mesh of polygons generated from these posts. The third is a hybrid of the previous two: it is a grid which has been generated by interpolating post values from a polygon mesh, which has in turn been generated from the SSDB posts. A GTDB can contain one or more of these three representations.

40.2.2.4 **Vertex Normals.** Some polygon-based image generators require the provision of normal vectors at each vertex in the polygonized data set. This boolean parameter specifies whether or not these should be generated during the polygonization process.

40.2.2.5 **Synthetic Culture.** Certain information in the SSDB is not derived from real-world source material, but is created by algorithmic means. This information may be either included or excluded from a GTDB, depending on the user's needs.

40.2.2.6 **Boundary SLOD Matching.** In GTDBs containing multiple Simulator Levels of Detail (SLODs) which are not geographically concident, there can be mismatches at the boundaries between SLODs, resulting in disconcerting anomalies in the terrain model. Specification of this parameter forces the transformation process to make the terrain models at different SLODs match up correctly, to avoid such effects.

40.2.2.7 **Convex Polygons.** This boolean parameter, when set to true, forces the transformation process to fragment model and culture polygons to eliminate convex edges.

40.2.2.8 **Edge Limit.** This parameter allows the user to specify the maximum number of edges permissible in a single polygon.

40.2.2.9 **Model References.** The user can specify the substitution of model references for culture features in the GTDB being generated. This results in a connecting link between the culture and model libraries, which are completely independent in the SSDB.

40.2.2.10 **Expanded Lineals.** If desired, the user can request that lineal features in the culture data set be replaced with polygons, the widths of which are determined by the "width" attributes of the SSDB lineal features.

40.2.2.11 **Fragmented Point Light Strings.** Many light features in the SSDB are not stored as individual features, but as components of strings containing N identical lights. If specified, this parameter allows the user to request that such features be broken into N individual light features, each independently stored and attributed.

40.2.2.12 **Face Count Exception.** The transformation from SSDB into GTDB will sometimes generate a conflict between a polygon count limit and a requirement for mandatory features. In the event that these two goals conflict, the user can specify one of three courses of action: ignore the limit on face count (which will yield a GTDB having more faces than the simulator can potentially handle), delete some of the features which were previously identified as being mandatory (resulting in the omission of some potentially important cues), or simply give up and abort the transformation run.

40.2.2.13 **SLOD Parameters.** Within a GTDB, there can be any number of user-defined SLODs. The following parameters are specified for each individual SLOD.

40.2.2.13.1 **Keep-List.** This identifies all of those features which are to be retained during the transformation from SSDB to GTDB. This list is necessary in those cases for which data base culling or filtering is anticipated, in order to retain those features which are important for a specific application. If a keep-list is not specified, features will be eliminated at random in order to avoid exceeding the specified face-count limit.

40.2.2.13.2 **Delete-List.** This is analogous to the keep-list, but with the opposite function; it identifies those features which are <u>not</u> wanted in the GTDB, and so these features are always deleted.

40.2.2.13.3 **Level-List.** This identifies those features which are required to be situated on level terrain, such as lakes; this parameter is used to force the terrain polygonization algorithm to flatten the terrain model under such features.

40.2.2.13.4 **Thinning Tolerance.** Often, features contain too many vertices for an image generator to handle. (This problem arises from the fact that, when a feature is broken down into convex polygons, the number of polygons which result is directly proportional to the number of vertices in the feature.) Thus, it is usually necessary for the transformation process to thin, or remove points from, a complex feature. This parameter specifies the amount of thining permissible in a given GTDB, as a function of the error it introduces into the shape of the resulting feature.

40.2.2.13.5 **Highest Level of Detail.** This is used to map SSDB LODs into GTDB SLODs. The SLOD will contain features extracted from SSDB levels up to and including the level specified.

40.2.2.14 **Area Block Parameters.** Within each SLOD, the data is comprised of a set of tiles or Area Blocks (ABs). Certain parameters can be specified for each individual area block within a SLOD, to allow for a non-homogeneous data base, even below the SLOD level. This can be used to concentrate the bulk of the transformation processing in only those areas of high interest, such as targets and corridors. The parameters which can be specified for each area block are described in the following subparagraphs.

40.2.2.14.1 **Area Block Dimensions.** This is used to specify the size of each individual area block, in 1/1000 arc-second increments up to 15 arc-minutes square. Adjacent area blocks must have identical corner coordinates on the shared edge.

40.2.2.14.2 **Face Count Limit.** If specified, this gives a maximum limit to the number of faces which can be generated for an area block.

40.2.2.14.3 **Model Reference Limit.** As with the face count, this limits the number of faces which can be generated for an area block.

40.2.2.14.4 **Terrain Goodness-of-Fit.** This parameter applies only to those GTDBs with polygonized terrain. It defines the maximum allowable deviation from the original terrain data from which the polygons are generated. In effect, this drives the number of terrain polygons generated; the tighter the tolerance, the more polygons. It should be noted, also, that the Project 2851 terrain polygonization algorithm generated irregularly-shaped triangles, and their density varies as a function of the terrain roughness.

40.2.2.14.5 **Number of Terrain Polygons.** This parameter can be used to specify the minimum and/or maximum number of terrain polygons to be generated for the area block. If it should conflict with the goodness-of-fit specified, the maximum number of terrain polygons discussed in paragraph 40.2.2.14.4 shall take precedence.

40.2.2.15 **Additional Islands.** A GTDB can contain any number of user-defined "islands", which are regions of nested SLODs. The coordinates of these can be specified independently.

40.3 **Data Formats.** Self-explanatory.

40.3.1 **Non-Texture Data.** Self-explanatory.

40.3.1.1 **Field Format.** Self-explanatory.

40.3.1.2 **Inter-Field Separation.** Self-explanatory.

40.3.2 **Texture Data.** Self-explanatory.

50 DETAILED REQUIREMENTS

(The information provided herein serves as a guide to the interpretation of requirements specified in Section 5 of this standard.)

50.1 Data Base Structure.

50.1.1 The following subparagraphs describe the detailed logical record and field structure of the GTDB files, as they would appear on a GTDB tape created by the Project 2851 software.

50.1.1.1 Data units called 'pseudo-files' have been defined for collections of records which should logically be treated as independent files but which, for performance reasons, have been physically grouped with other pseudo-files within a larger physical file. In earlier versions of the GTDB design, the proliferation of relatively small physical files was found to cause highly inefficient use of the magnetic tape medium (due to the inter-file gaps), as well as inordinate I/O processing overhead. The grouping of pseudo-files into fewer physical files has been found to improve I/O performance dramatically.

50.1.1.2 Data 'fields' are logical data units occurring at predefined locations within a record. Fields maintained internally as non-ASCII data (such as integers, floating point, Boolean, and Ada enumerated types) will be converted to ASCII during GTDB generation. The size of each ASCII field is fixed and will contain enough character positions to hold the largest valid value for the field, including a sign for numerics. Adjacent fields are separated by NULL characters to permit simple extraction of individual fields based on relative field position without consideration for the length of intervening fields. Fields may be made up of sub-fields, which are also delimited by NULL characters. The format and meaning of individual fields, as well as default values, are defined in Appendix A to this standard.

50.1.1.3 Texture image data will not be converted to ASCII. This data will be either in its original image file format or in the NITF file format, depending on the type of texture requested. All original source data will follow its own formatting conventions. All NITF data will follow NITF formatting conventions. NITF files use carriage-return and line-feed characters as inter-field separators and a blank character as inter-item separator.

50.1.2 Section 5 of this standard describes the architecture of the GTDB. Using the nomenclature specified in DI-MCCR-80028, the data base is described as a set of 'files', which in turn consist of a set of 'records', which in turn consist of a set of 'fields', which may further consist of a set of 'items'.

50.1.2.1 There may be as many as 16 distinct physical files within a GTDB. These files form a logical hierarchy of pointers and information.

50.1.2.2 Starting at the top, a GTDB begins with a Gaming Area Header File which describes the general characteristics of the entire data base. A gaming area is simply the geographic area of data base coverage over which simulated operations are to be supported. If a radar GTDB is being concurrently generated with a visual GTDB, their gaming areas are not required to coincide (or even, from a software standpoint, to relate to each other in any way). For example, it would be possible to specify a larger gaming area for the radar GTDB to reflect the greater sensor range.

50.1.2.3 At the next level of the hierarchy, there are header files describing and pointing to three major classes of data within the GTDB. The Simulator Level of Detail (SLOD) Header File and the Area Block Header File point to terrain and culture data; the Model Library Header File points to model libraries; and the Texture Library Header File points to texture libraries.

50.1.2.4 SLOD Header Files define the general characteristics of each of the SLODs defined for the particular GTDB. A SLOD is a distinct level within a range of levels of data base terrain/culture content density and/or complexity. Different SLODs may be used to simulate variations in proximity, sensor range, sensor gain, and field of view. Typically, each SLOD will cover the entire gaming area, but this is not required by the GTDB or CDBTP.

50.1.2.5 It is also not required that the data density be consistent throughout a SLOD. Within each SLOD, the CDBTP will allow the user to specify a hierarchy of "islands" of coverage within which the culture data may be of higher detail than surrounding areas. Different levels of terrain detail may be specified on an area block basis. (See the Software Detailed Design Document (SDDD) for the CDBTP for a detailed description of SLOD specification options and constraints.) The CDBTP is parameterized to permit highly tailored definitions of the number and content of SLODs within any given GTDB.

50.1.2.6 Within each simulator level of detail, the area covered by the gaming area may be subdivided into rectangular area blocks, defined by arcs of latitude and longitude. The area blocks are the units of data dynamically loaded and transformed by a real-time simulator. Due to differences in data density between different Simulator Levels of Detail (SLODs), the area block size will normally vary from simulator level of detail to simulator level of detail within a GTDB. Typically, the area block size will be constant within a simulator level of detail, but this is not a restriction of the GTDB or common data base transformation program. The only restrictions imposed by the CDBTP are that a side of an area block may not exceed 15 minutes of arc, and that the shared boundaries between neighboring area blocks must be mutually identical (i.e., an area block cannot share a boundary with two smaller area blocks).

50.1.2.7 The terrain and culture data within any given area block are categorized and stored in as many as nine types of area block pseudo-files. These are shown in Figure C-1. Two of the pseudo-files are mandatory in every area block. The Areal Feature Area Block (AFAB) Pseudo-File will always be present because, by convention, there will always be at least one areal feature in any area block (i.e., the background feature). The Vertex Area Block (VAB) Pseudo-File contains the vertex coordinates used to define the spatial boundaries of features (as well as terrain, when polygonized) and, hence, will always be present to define the background feature at a minimum. The presence of the other pseudo-files in a particular GTDB (or in any particular simulator level of detail and area block of the GTDB) is optional, depending on the availability of data and on the requirements of the target simulator and training mission. The Area Block Header File identifies those lower-level area block pseudo-files which are present for any given area block within a SLOD.

Figure C-1.

50.1.2.8 Terrain data can be represented within the GTDB either as a continuous network of surface polygons in three dimensions, or as a systematically spaced grid matrix of elevation posts. The user has the option of requesting one or the other representation, or both within the same GTDB, or neither if terrain is not desired.

50.1.2.9 If gridded terrain is requested, the user may select the terrain post density from available LODs stored in the Standard Simulator Data Base (SSDB). The standard simulator data base has been designed to store terrain data, if available, at nominal post spacings of 100m (3 arc seconds of latitude/longitude), 30m (1 arc second), 10m (0.3 arc seconds), and 1m (0.03 arc seconds).

50.1.2.10 If polygons are requested, the user may specify common data base transformation program transformation parameters controlling the trade-off between polygon budgets and a goodness-of-fit criterion. Project 2851 has selected the public-domain Dirichlet tesselation and Delauney polygonization algorithms to achieve maximum terrain fidelity and correlation capability, given variable polygon budget constraints. These algorithms produce variable-density triangles, allocated to provide more polygons in areas of relatively greater terrain roughness. (Refer to the common data base transformation program software detailed design document for details on these algorithms.) Although the common data base transformation program generates only triangles, the GTDB data structure could support more complex polygons.

50.1.2.11 If correlation is desired between a GTDB using polygonized terrain and another (or the same) GTDB using gridded terrain, a user-specified option will cause the common data base transformation program to derive equivalent gridded post values from the polygonized terrain rather than use the originally compiled standard simulator data base values.

50.1.2.12 Culture data will include both spatial and descriptive information about individual features which lie upon the terrain. Both natural and man-made features are included. The general data architecture for culture data is an expansion of standards and conventions established by the Defense Mapping Agency (DMA). Spatially, features will be represented as discrete points, lines, or surface polygons in two-dimensional space. The height of an object above the terrain (if applicable) is encoded as an attribute of the feature along with many other attributes useful for generating simulated scenes. Some of the attributes have obvious direct value for scene depiction, e.g., reflectance, directivity, and Surface Material Category (SMC). Attributes such as percent of tree coverage are required to predict radar masking effects. Other attributes (e.g., Feature Descriptor Code, Roof Type) provide semantic detail about features which may be helpful to simulators in selecting models, texture, or other means for realistic rendering of culture objects. Still other attributes (e.g., maximum load-bearing capacity of a bridge, temporal characteristics) may be useful to support realistic training scenario content.

50.1.2.13 As noted above, the spatial coordinates describing the boundary vertices of a feature will normally be in two-dimensional (x,y) space, as they are stored in the SSDB. Selection of a common data base transformation program option for fragmenting culture along the boundaries of underlying terrain polygons will result in culture vertices being placed in 3-D (x,y,z) space, such that all features will be coplanar with the underlying terrain.

50.1.2.14 As is the case with terrain, culture data may be selected from the various Levels of Detail (LODs) of the Standard Simulator Data Base (SSDB), as data are available. Culture LODs have been defined in the standard simulator data base to correspond roughly to feature resolutions of 300m, 100m, 30m, 10m, 3m, and 1m. The previously described "island" specifications are used by the common data base transformation program to allocate culture detail where it is most needed for the training application.

50.1.2.15 If correlation between a radar GTDB and a visual GTDB is desired, features typically not associated with radar data bases (e.g., models and texture) may optionally be included in the radar GTDB, so that radar effects can be correlated with visual/infrared scenes. On the other hand, the need to correlate with a visual system can have a limiting effect on the radar GTDB. Typically, the computational demands on visual image generators are significantly greater than on radar image generators. As a result, visual systems tend to be more limited in the number of features or surfaces they can portray compared to radar systems. This means that when correlation is a requirement, some radar-significant features may have to be eliminated from the radar scene.

50.1.2.16 The Project 2851 GTDBs have a provision for carrying correlation priority codes indicating which features are more or less important to retain for correlation. When populated, these priority codes may be used to guide the filtering process whenever features have to be eliminated to meet image generator processing constraints.

50.1.2.17 The model libraries contain the identification and descriptive information for computer-graphic models of various generic (e.g., a 'typical' house) and specific (e.g., a C-130 transport) objects which must be displayed with more realism and detail than is possible with simple 2-D surface culture polygons. It should be emphasized that all models in a GTDB will be polygonal (surface-based) models, even though models are represented in the standard simulator data base using Constructive Solid Geometry (CSG). Together with the CSG definition of the basic model, the standard simulator data base will store instructions for automatically converting the single generic model into various polygonal representations more suitable for use on particular real-time image generators. These instructions are executed by the Common Data Base Transformation Program (CDBTP) when selecting models for inclusion in a GTDB.

50.1.2.18 Like terrain and culture, models may be described at multiple levels of detail to allow trade-offs between rendering detail and system performance considerations. The model library will initially contain three levels of detail, designated as high, medium, and low. Medium-detail models will define surfaces and components at roughly 1-meter resolution and are intended for nominal viewing ranges. High-detail models are intended for close-up viewing and will resolve components at sub-meter resolutions. Low-detail models are intended for far-scene or low-fidelity displays and will always consist of 20 or fewer polygons, regardless of component resolution.

50.1.2.19 Each of the model libraries has associated with it a model vertex file, containing a list of coordinate vertices defining the geometry of the models. Each model vertex file is organized as a set of pseudo-files, one per model. Each pseudo-file contains the vertices used to define all LODs of a model.

50.1.2.20 Models introduce further problems in the area of correlation between GTDBs. The concept is to supply generic models which will support correlation by providing a common geometry, or at least reasonably compatible geometries, across sensors and LODs. Since models tend to be highly specific to the vendor's architecture and also are a major focus of product differentiation and improvement, it may not be practical or desirable to force vendors to always use the generic models. However, the generic models may take precedence when correlation is a higher priority than absolute rendering quality.

50.1.2.21 The texture libraries contain descriptive information and digital image files which may be applied to a terrain/culture or model surface to gain greatly increased realism and detail without increasing the complexity of the underlying geometry. Two texture libraries are supported: areal and model. Both of these libraries support supplying textures processed at a variety of stages and resolutions, a choice of specific or generic texture, and texture types including color and grayscale. Areal texture libraries support an additional texture type of SMC/FDC codes for non-visual simulation (e.g., radar or infra-red). Naturally, areal texture applies to terrain and culture, while model texture is used only for models.

50.1.2.21.1 Specific texture is specific to a certain geographic area for areal texture and specific to a certain object for model texture. Specific texture is an image generally derived from satellite imagery, aerial photography, or hand-held ground photography. Generic texture is an image that can be generated from specific texture or from synthetic pattern generation. Generic texture is a small tile of texture that can be repeated to simulate the true view without using the large amounts of memory and I/O processing necessary with specific texture. An example of areal generic texture would be a field pattern that is repeated over and over again to achieve the view of a very large field. An example of model generic texture would be a brick wall pattern that could be repeated along the side of a building.

50.1.2.21.2 To accommodate the many needs of the simulation community, the GTDB supports texture processed at any one of five stages. Stage 1 texture is the original digital image provided by the source. The texture is in its original native format as that provided by the source. The GTDB will provide with it descriptive information such as ground control points, tie points to other images, percentage of cloud cover, geographic location, and image capture date and time. Naturally, this stage applies only to specific texture. Stage 1 processing is available for both areal and model texture.

50.1.2.21.3 Unlike Stage 1 texture, Stage 2 texture is provided in National Imagery Transmission Format (NITF) (Version 1.1, 1 March 1989, and NITF Change Notice No. 2, 23 May 1990). This is a DOD standard for images. The content of the image may have changed due to the "clean up" operations performed at this stage. Noise removal, occlusion removal with hole fill-in, shadow minimization, haze removal, and contrast enhancement are examples of processing done to produce a Stage 2 texture. Once again, Stage 2 processing applies only to specific texture. Since this is the first stage where the original image's source format is not used, many of the GTDB texture parameters apply here. For instance, the user can specify, the number of bits per texture pixel (texel), and a texture format (Band-Sequential, or Band-Interleaved). The texture image and all descriptive information is supplied in NITF. It is available for both areal and model texture. The NITF standard is used at this stage.

50.1.2.21.4 Stage 3 texture is geometrically corrected, i.e., it is geopositioned and orthorectified. Areal texture is placed in equal arc spacing at this stage, while model texture is placed in a local Cartesian coordinate system. At this stage, image-to-image radiometric corrections have been performed. Most GTDB parameters apply at this stage. Stage 3 can apply to both specific and generic texture. It is available for both areal and model texture.

50.1.2.21.5 Stage 4 texture is simply Stage 3 texture transformed to a user-specified coordinate system other than geodetic for geographic areal texture (which is supplied in Stage 3). The NITF standard is used at this stage. This stage applies to both specific and generic texture. It is used only for areal texture, since model texture is always in a local Cartesian coordinate system that need not be transformed.

50.1.2.21.6 Finally, Stage 5 texture includes polygon mapping control information so that the texture is already mapped to the polygons. Thus, Stage 5 consists of Stage 3 or Stage 4 textures with this control information. Specific areal textures are mapped to terrain polygons through vertex-to-vertex mapping. Generic areal textures are mapped to culture polygons using global-based and face-based mapping. Specific model textures are mapped to model polygons using the vertex-to-vertex, face-based, and model-based mapping schemes. Generic model textures are mapped to model polygons using the face-based and model-based mapping methods. Once again, the NITF standard is used. Stage 5 processing can be done with either specific or generic texture.

50.1.2.21.7 For non-visual simulation systems, the GTDB supports texture maps containing a grid of Surface Material Categories (SMC) and Feature Descriptor Codes (FDC). Such a map provides a finer resolution of surface information than can be provided by vector data bases. These SMC/FDC textures are provided in the Areal Texture Library when requested. They are treated as specific areal textures. They can be produced at Stages 3, 4, or 5.

50.1.2.21.8 Textures are linked to terrain/culture in a number of ways. First, the Area Block Header File may have a table of pointers to specific textures for each area block. These textures can be at any of Stages 1 through 4 of processing. Second, each terrain polygon within an area block may have texture references for the actual texture mapping. This method would be used for Stage 5 texture mapping of specific areal texture. This data can be found in the Terrain Polygon Area Block Pseudo-File. Third, for geographic areas where specific areal texture is not available, generic texture can be used. Each areal feature within the Areal Feature Area Block Pseudo-File may have a pointer to a generic texture pattern that can be used in lieu of specific texture. Fourth, the Areal Feature Area Block Pseudo-File may also have the mapping information for Stage 5 generic textures.

50.1.2.21.9 Textures are linked to models in a number of ways. Models have texture references for non-mapped textures. This would pertain to Stages 1, 2, and 3. Stage 4 does not apply to models. Models have three other types of texture references, each for texture mapping at Stage 5. Vertex-to-vertex, face-based, and model-based mapping schemes all support specific texture while only the latter two can be used for generic texture. All of this data can be found in the Model Library Files. Under this design, the GTDB offers much flexibility to the user in the ways that texture can be requested for customized usage.

50.1.2.22 The design for the Project 2851 System permits a high degree of tailoring of GTDBs to match a specific simulator and training mission. For this reason, various combinations of data files described in this data base design document are possible. For example, it will be possible to request a GTDB in which all linear features have been converted to areals, eliminating the need for a Linear Feature Area Block File. In addition, the specific contents of the files may also vary considerably based on the transformation parameters specified. For example, a highly restrictive maximum polygon density would result in features being left out of a GTDB that would otherwise be represented. Under any circumstance, all GTDBs will conform to the logical architecture as specified herein.

50.1.3 The following subsections describe each of the GTDB files. Note that the subsection numbers correspond to those of Section 5 of this standard, for easy reference.

50.2 **Gaming Area Header (GAH) File.** The purposes of the GAH are to: (1) identify the GTDB; (2) describe general gaming area characteristics; and (3) document the Project 2851 parameters used to generate the GTDB. GTDB identifiers will include a Project 2851 catalog number, date of creation, and security level. Gaming area descriptors will include coordinates specifying the geographic boundaries of the gaming area. GTDB production parameters will document the Project 2851 transformation parameters specified to create this particular GTDB. Specific parameter options are documented in the design of the Common Data Base Transformation Program (CDBTP).

50.2.1 **GAH Record Order.** Self-explanatory.

50.2.2 **GAH Field Structure.** Self-explanatory.

50.2.2.1 **GAH Identifier Record.** Self-explanatory.

50.2.2.2 **GAH File Name Record.** Self-explanatory.

50.2.2.3 **Gaming Area Header Record.** Self-explanatory.

50.2.2.4 **GTDB Parameters Record.** This record is mandatory and contains a list of all Common Data Base Transformation Program (CDBTP) transformation parameters used to generate this particular GTDB. (See Appendix A, Volume I, to the CDBTP software detailed design document for specifications of the meaning of the individual parameters.)

50.2.2.4.1 **Coordinate System Parameters Subrecord.** This subrecord contains GTDB parameters defining the coordinate system in which all terrain and culture spatial data are expressed. The parameters specifically define control information used to perform coordinate conversions, projection transformations, and/or datum shifts required to transform geodetically expressed standard simulator data base data into a user-specified GTDB coordinate space.

50.2.2.4.2 **Specific Areal Texture Parameters Subrecord.** This subrecord contains GTDB parameters used to generate a particular set of specific areal textures for a GTDB. Other parameters dealing with specific areal texture are at the Area Block level within the GAH.

50.2.2.4.3 **Generic Areal Texture Parameters Subrecord.** This subrecord contains GTDB parameters used to generate a particular set of generic areal textures for a GTDB. Other parameters dealing with generic areal texture are at the SLOD level within the GAH.

50.2.2.4.4 **Specific Model Texture Parameters Subrecord.** This subrecord contains GTDB parameters used to generate a particular set of specific model textures for a GTDB. These parameters apply to all models referenced by culture. Texture parameters for explicitly requested models are provided at the model level. All available model texture resolutions are automatically provided.

50.2.2.4.5 **Generic Model Texture Parameters Subrecord.** This subrecord contains GTDB parameters used to generate a particular set of generic model textures for a GTDB. These parameters apply to all models referenced by culture. Texture parameters for explicitly requested models are provided at the model level. All available model texture resolutions are automatically provided.

50.2.2.4.6 **Time of Year Preference Subrecord.** Self-explanatory.

50.2.2.4.7 **Image Capture Data Range Subrecord.** Self-explanatory.

50.2.2.4.8 **Acceptable Percentage of Cloud Cover Subrecord.** Self-explanatory.

50.2.2.4.9 **Acceptable Percentage of Shadow Cover Subrecord.** Self-explanatory.

50.2.2.5 **Boundary Point Record.** This record contains a geographic coordinate used to define a point on the boundary of the gaming area covered by the GTDB. The arc between each successive pair of points defines an edge of the gaming area. By software detailed design document convention, the shape of a gaming area may be irregular, but each edge must be parallel to one of the axes of the coordinate plane. The gaming area boundary will be closed explicitly; i.e., the first boundary point will be explicitly listed again as the last point. Thus, there will always be at least five Boundary Point Records. The boundary point records for a gaming area will be sequenced in counterclockwise order as viewed from above.

50.2.2.6 **Model List Record.** This optional record contains the ID of a model which has been explicitly requested for inclusion in the GTDB. The model list is intended to be used to select models (e.g., aircraft) in addition to any used for culture substitution. (Models which are referenced within the Model Reference Area Block File as substitutes for culture will automatically be included in the GTDB model libraries, whether or not they appear in the model list.)

50.2.2.6.1 **Specific Model Texture Parameters Subrecord.** This subrecord contains GTDB parameters used to generate a particular set of specific model textures for an explicitly specified model within the GTDB. All available model texture resolutions are automatically provided.

50.2.2.6.2 **Generic Model Texture Parameters Subrecord.** This subrecord contains GTDB parameters used to generate a particular set of generic model textures for an explicitly specified model within the GTDB. All available model texture resolutions are automatically provided

50.2.2.7 **SLOD Parameters Record.** This record and its subsidiary records contain the common data base transformation program transformation parameters used to generate a particular Simulator Level of Detail (SLOD) within the GTDB. There will always be at least one simulator level of detail in a GTDB.

50.2.2.8 **Keep-List Record.** This optional record contains starting and ending values defining a range of Project 2851 Feature Descriptor Codes (FDCs). All standard simulator data base features whose FDCs fall within the given range are candidates for inclusion in the GTDB. The Project 2851 FDCs are extensions of DMA's FACS (Feature Attribute Coding Standard) feature codes.

50.2.2.9 **Delete-List Record.** This optional record contains starting and ending values defining a range of Project 2851 Feature Descriptor Codes (FDCs). All standard simulator data base features whose FDCs fall within the given range will be excluded from the GTDB. The Project 2851 FDCs are extensions of DMA's FACS (Feature Attribute Coding Standard) feature codes.

50.2.2.10 **Level-List Record.** This optional record contains starting and ending values defining a range of Project 2851 Feature Descriptor Codes (FDCs). All standard simulator data base features whose FDCs fall within the given range will be used to level (flatten) the underlying terrain in the GTDB. The Project 2851 FDCs are extensions of DMA's FACS (Feature Attribute Coding Standard) feature codes.

50.2.2.11 **Area Block Parameters Record.** This record contains the common data base transformation program transformation parameters used to generate a particular Area Block within a simulator level of detail of a GTDB. There will always be at least one area block per simulator level of detail in a GTDB. Texture parameters are specified here for specific texture.

50.2.2.12 **Island Record.** This record is used to define the existence of an "island" within the GTDB. An island is an area within the gaming area (or within a larger island) having assigned levels of culture resolution at various SLODs. Islands are used to control the allocation of culture detail so that high-interest areas (e.g., airfields or targets) can be modeled with much greater scene detail than areas of lesser importance. Figure C-2 illustrates the island concept. By convention, the entire gaming area constitutes the first, largest island. Any subsequent islands will define sub-areas of previously-defined islands. The common data base transformation program will not allow island boundaries to overlap.

50.2.2.13 **Island LOD Record.** This record is used to define the levels of culture resolution to be applied within an island at the various SLODs. This record specifies which culture LOD within the standard simulator data base is to be used to extract data for inclusion within this island at the given SLOD. Texture parameters are specified here for generic texture.

Figure C-2

50.2.2.14 **Island Boundary Point Record.** This record contains a geographic coordinate used to define a point on the boundary of an island within the gaming area covered by the GTDB. The arc between each successive pair of points defines an edge of the island. By common data base transformation program convention, the shape of an island may be an irregular closed contour. Unlike gaming area boundaries, island boundaries do not require that each edge be parallel to one of the axes of the coordinate plane. Each island will be closed explicitly; i.e., the first boundary point of the island will be explicitly listed again as the last point. Thus, there will always be at least three Island Boundary Point Records per island. The boundary point records for each island will be sequenced in counterclockwise order as viewed from above.

50.2.2.15 **Option Record.** This record contains an option value which indicates whether the GTDB tape contains all area blocks or just those added or updated since the previous GTDB version.

50.2.2.16 **Affected AB Count Record.** This record contains a count of the area blocks added or updated since the previous GTDB version.

50.2.2.17 **Affected AB ID Record.** This record identifies an area block added or updated since the previous GTDB version.

50.2.2.18 **Checksum Record.** This record contains the checksum value for the Gaming Area Header File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the file. The checksum does not include those characters contained in the Checksum record itself.

50.3 **Model Library Header (MLH) File.** The Model Library Header (MLH) File is mandatory in a GTDB. The MLH identifies model libraries (if any) contained within the GTDB. The MLH will exist whether or not a GTDB contains any model libraries. The purposes of the MLH are to: (a) identify any model libraries which do exist; and (b) provide data base statistics useful for planning and decision making during use of the model libraries. Model libraries will be identified as a 2-D Static Model Library, a 3-D Static Model Library, or a 3-D Dynamic Model Library. Each of these libraries is optional and will exist only when there are models of the given type being passed in the GTDB. There will be a maximum of one of each type of model library in a GTDB. Data base statistics will give user-created Formatter Programs (and human data base modelers) a means for planning and optimizing the selection of models for the target simulator. The types of statistics to be provided will include model-specific measures of polygon/face complexity and color distribution.

50.3.1 **MLH Record Order.** Self-explanatory.

50.3.2 **MLH Field Structure.** Self-explanatory.

50.3.2.1 **MLH Identifier Record.** Self-explanatory.

50.3.2.2 **File Name Record.** Self-explanatory.

50.3.2.3 **Model Library Header Record.** The "Model Library Type" field indicates which of the three libraries is being described. A value of zero in the "Number of Models" field indicates that there is no actual model library file of the given type within the GTDB.

50.3.2.4 **Culture Color Table Record.** This record contains a single element in a distribution table of colors used in the model library.

50.3.2.5 **Light Color Table Record.** This record contains a single element in a distribution table of light-emitter colors used in the model library.

50.3.2.6 **Model LOD Complexity Table Record.** This record contains a count of the number of LODs that exist for a given model in the model library.

50.3.2.7 **Model Complexity Statistics Table Record.** This record contains complexity statistics for each LOD version of a model.

50.3.2.8 **Checksum Record.** This record contains the checksum value for the Model Library Header File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the file. The checksum does not include those characters contained in the Checksum record itself.

50.4 **Simulator Level of Detail Header (SLODH) File.** The Simulator Level of Detail Header (SLODH) File is mandatory in a GTDB. The SLODH identifies all of the SLODs contained within the GTDB. The purposes of the SLODH are to: (a) identify the SLODs contained within the data base; and (b) provide data base statistics useful for planning and decision making during use. A minimum of one SLOD is required for a valid GTDB, but a typical GTDB is expected to have several SLODs. A SLOD is a distinct level of data base detail and/or density. Different SLODs may be used to simulate variations in proximity, sensor range, sensor gain, and field of view. Each simulator level of detail may be logically subdivided into area blocks as defined in the Area Block Header File. Data base statistics will give user-created Formatter Programs (and human data base modelers) a means for planning and optimizing the selection and filtering of SLODs for the target simulator. The types of statistics to be provided include measures of feature and polygon density, feature type distribution, SMC distribution, and color distribution.

50.4.1 **SLODH Record Order.** Self-explanatory.

50.4.2 **SLODH Field Structure.** Self-explanatory.

50.4.2.1 **SLODH Identifier Record.** Self-explanatory.

50.4.2.2 **File Name Record.** Self-explanatory.

50.4.2.3 **SLODH File Header Record.** Self-explanatory.

50.4.2.4 **SLOD Header Record.** This record contains control information describing a particular SLOD.

50.4.2.4.1 **SLOD Polygon Density Statistics Table Subrecord.** This subrecord contains statistics on maximum and minimum densities of data base elements on a per-area-block basis within the SLOD.

50.4.2.5 **Feature Distribution Table Record.** This record contains a single element in a distribution table of types of culture features present within a simulator level of detail, categorized and sorted by Feature Descriptor Code.

50.4.2.6 **Z-Density Distribution Table Record.** This record contains a single element in a distribution table of the numbers of layers of culture occurring above terrain polygons within a SLOD. These records will be omitted when the GTDB does not include polygonized terrain, or if the culture has not been fragmented on the terrain.

50.4.2.7 **SMC Distribution Record.** This record contains a single element in a distribution table of surface material codes present within a simulator level of detail, categorized and sorted by Surface Material Category and Surface Material Subtype.

50.4.2.8 **Culture Color Table Record.** This record contains a single element in a distribution table of colors used in the SLOD.

50.4.2.9 **Light Color Table Record.** This record contains a single element in a distribution table of light-emitter colors used in the SLOD.

50.4.2.10 **Checksum Record.** This record contains the checksum value for the Simulator Level of Detail Header File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the file. The checksum does not include those characters contained in the Checksum record itself.

50.5 **Area Block Header (ABH) File.** The Area Block Header (ABH) File is mandatory in a GTDB. The ABH identifies all area blocks contained within all SLODs in the gaming area. The purposes of the ABH are to: (a) identify the area blocks contained within the data base; and (b) provide data base statistics useful for planning and decision making during use. Area blocks are predefined rectangular subdivisions of the gaming area, defined by arcs of latitude and longitude. A minimum of one area block per simulator level of detail is required for a valid GTDB, but a typical GTDB is expected to have many area blocks per SLOD. Each area block will be identified by a unique ID code, as well as by two corner coordinates specifying the geographic boundaries of the block. These coordinate pairs define the southwest and northeast corners of the area block. Data base statistics will give user-created Formatter Programs (and human data base modelers) a means for planning and optimizing the selection and filtering of area blocks for the target simulator. The types of statistics to be provided include measures of feature and polygon density, feature type distribution, surface material distribution, color distribution, and terrain roughness. The ABH will contain flags indicating which of the possible set of terrain and culture area block data pseudo-files are present for this area block in the GTDB. These pseudo-files may include a Vertex Area Block (VAB) Pseudo-File, an Areal Feature Area Block (AFAB) Pseudo-File, a Linear Feature Area Block (LFAB) Pseudo-File, a Point Feature Area Block (PFAB) Pseudo-File, a Point Light Feature Area Block (PLFAB) Pseudo-File, a Point Light String Feature Area Block (PLSFAB) Pseudo-File, a Terrain Polygon Area Block (TPAB) Pseudo-File, a Terrain Grid Area Block (TGAB) Pseudo-File, and a Model Reference Area Block (MRAB) Pseudo-File. Only the VAB and the AFAB are required in every GTDB. The exact configuration of area blocks in any given GTDB will depend on the transformation parameters used, and on the availability of the various types of data within the SSDB. The area block data pseudo-files for all area blocks in the data base are contained in Simulator Level of Detail Area Blocks (SLAB) File(s), which will follow the photo texture and model library files.

50.5.1 **ABH Record Order.** Self-explanatory.

50.5.2 **ABH Field Structure.** Self-explanatory.

50.5.2.1 **ABH Identifier Record.** Self-explanatory.

50.5.2.2 **File Name Record.** Self-explanatory.

50.5.2.3 **ABH File Header Record.** Self-explanatory.

50.5.2.4 **Area Block Header Record.** The Area Block Header record contains control information describing a particular area block within a SLOD.

50.5.2.4.1 **Polygon Density Statistics Table Subrecord.** This subrecord contains counts of data base elements within the area block.

50.5.2.4.2 **Terrain Roughness Statistics Subrecord.** This subrecord contains statistics indicating the roughness (i.e., variability) of the terrain within the area block. These fields will be populated with real data whenever terrain (either gridded or polygonized, or both) has been requested.

50.5.2.4.3 **Area Block Existence Flags Subrecord.** This subrecord consists of a series of boolean flag fields indicating whether an area block of a particular type exists within this SLOD. Each flag will contain a "T" or "F" to indicate existence or non-existence, respectively.

50.5.2.5 **Feature Distribution Table Record.** This record contains a single element in a distribution table of types of culture features present within an area block, categorized and sorted by Feature Descriptor Code.

50.5.2.6 **SMC Distribution Record.** This record contains a single element in a distribution table of surface material codes present within an area block, categorized and sorted by Surface Material Category and Surface Material Subtype.

50.5.2.7 **Culture Color Table Record.** This record contains a single element in a distribution table of colors used in the area block.

50.5.2.8 **Light Color Table Record.** This record contains a single element in a distribution table of light-emitter colors used in the area block.

50.5.2.9 **Areal Texture Table Record.** This record contains a single element in a distribution table of areal textures used in the area block. The areal textures are sorted by Texture ID. This record can reference specific areal texture in any one of Stages 1 through 4 and SMC/FDC texture in either Stages 3 or 4. (There is no SMC/FDC texture in Stages 1 and 2.)

50.5.2.10 **Checksum Record.** This record contains the checksum value for the Area Block Header File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the file. The checksum does not include those characters contained in the Checksum record itself.

50.6 **Texture Library Header (TLH) File.** The Texture Library Header (TLH) File is mandatory in a GTDB. The TLH identifies photo texture libraries (if any) contained within the GTDB. The TLH will exist whether or not a GTDB contains any photo texture image libraries. The purposes of the TLH are to: (a) identify any photo texture libraries which do exist; and (b) provide data base statistics useful for planning and decision making during use of the texture libraries. Photo texture libraries will be identified as either an Areal Photo Texture Library or a Model Photo Texture Library. Each of these libraries is optional and will exist only when there are texture maps of the given type being passed in the GTDB. There will be a maximum of one of each type of photo texture library in a GTDB. Data base statistics will give user-created Formatter Programs (and human data base modelers) a means for planning and optimizing the selection of texture maps for the target simulator. The types of statistics to be provided will include the number of textures for each stage of specific, generic, and SMC/FDC textures, and storage size requirements for each of those classifications.

50.6.1 **TLH Record Order.** Self-explanatory.

50.6.2 **TLH Field Structure.** Self-explanatory.

50.6.2.1 **TLH Identifier Record.** Self-explanatory.

50.6.2.2 **File Name Record.** Self-explanatory.

50.6.2.3 **Texture Library Complexity Statistics Record.** Self-explanatory.

50.6.2.4 **Texture Library Header Record.** The "Texture Library Type" field indicates which of the two libraries is being described. A value of zero in the "Number of Texture Images in Library" field indicates that there is no actual texture library file of the given type within the GTDB.

50.6.2.5 **Texture Distribution Table Record.** This record contains control data for a texture image in the texture library. This record contains control data for a texture image in the texture library. The number of these records will correspond to the value in the "Number of Texture Images in Library" field in the parent Texture Library Complexity Statistics record. Thus, there is a set of these records for each texture library. The "Texture Data Format" field shall contain the value "NITF" for all textures other those in Stage 1; for those Stage 1 textures, the field shall accurately describe the source format. The "Number of Data Files" field value shall always be 2 for a non-Stage 1 texture: the NITF Image Sub-Header File and the NITF Image Data File. For Stage 1 textures, the "Number of Data Files" field value shall be 2 or greater, depending on the number of Original Format Image Files from the source. A Stage 1 texture shall always have an NITF Image Sub-Header File to include P2851-specific data.

50.6.2.6 **Stage 1 Texture Field Association Record.** This optional record contains a Stage 1 texture file name and its associated original file name. All GTDB texture file names can be automatically derived; however, for Stage 1 texture data, a file's original name assigned by the source may have value to a GTDB user. This name is associated with the current GTDB texture file name here for the user's convenience.

50.6.2.7 **Checksum Record.** This record contains the checksum value for the Texture Library Header File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the file. The checksum does not include those characters contained in the Checksum record itself.

50.7 **2-D Static Model (2DSM) Library File.** The 2DSM is optional in a GTDB. When present, it contains all the 2-D static models referenced within the GTDB or explicitly requested during GTDB generation. The geometry of each model is represented as a set of surface polygons in 2-D space. The perimeter of each polygon is described by a set of coordinates or vertices. Each set of coordinates is identified by its position in a list of values stored in a separate 2-D Static Model Vertex File, allowing each coordinate set to be used repeatedly. By convention, all GTDB model polygons will be closed implicitly rather than explicitly; i.e., the first vertex of a polygon will not be explicitly listed again as the last vertex. Each surface or polygon may have descriptive and rendering attributes associated with it. A wide range of fields are available to describe attributes specific to radar, visual, and infrared simulation, as well as general attributes applicable to all three. Each polygon may reference any number of photo texture maps from the Model Photo Texture Library. The model library structure also supports composite models in which one model references another as a component. The 2DSM supports storage of models for any given object at multiple levels of detail. This will permit selection of models at a level of complexity that balances the desire for realism and the resolution of the simulated sensor with the processing capacity of the image generator and the overall complexity of the scene.

50.7.1 **2DSM Record Order.** Self-explanatory.

50.7.2 **2DSM Field Structure.** Self-explanatory.

50.7.2.1 **2DSM Identifier Record.** Self-explanatory.

50.7.2.2 **File Name Record.** Self-explanatory.

50.7.2.3 **2DSM Header Record.** This mandatory record contains control information describing this file.

50.7.2.4 **Model Header Record.** This record identifies a model group, which can consist of one or more actual model geometries representing a given object at varying LODs.

50.7.2.5 **LOD Header Record.** This record describes a particular LOD version of a model. (Fields designated "Always Zero" are not applicable to 2-D static models. They have been included for consistency of format across all model libraries.) The Placement Point Field will be supported through the use of FACS codes.

50.7.2.6 **LOD Texture Reference Pointer Record.** This record provides pointers to the texture references for a model LOD.

50.7.2.7 **Component Header Record.** This record describes a particular component within a model LOD.

50.7.2.8 **Component Texture Reference Pointer Record.** This record provides pointers to the texture references for a component.

50.7.2.9 **Model Polygon Record.** This record describes the attributes of a polygon within a particular model. The last polygon record will always define the model "footprint" as referenced by the Base Polygon ID field in the parent Model record. This polygon is not meant to be displayed during a simulation but is supplied to aid the user in placing additional instances of the model.

50.7.2.10 **Model Microdescriptor Record.** This optional record contains a microdescriptor associated with a model polygon. The types of microdescriptors supported by Project 2851 are listed in Section 5.16 of this standard.

50.7.2.11 **Vertex Pointer Record.** This record is used to associate a model polygon with a vertex record within the Two-Dimensional Static Model Vertex (2DSMV) File. There will be three or more of these records defining the geometry of each model polygon. By convention, a model polygon will be closed implicitly rather than explicitly; i.e., the first vertex of a polygon will not be explicitly referenced again as the last vertex.

50.7.2.12 **Polygon FACS Record.** This optional record contains a FACS (Feature Attribute Coding Standard) value associated with a model polygon.

50.7.2.13 **Polygon Texture Reference Record.** This optional record is used to associate a model polygon with a Photo Texture Image Header record within the Model Photo Texture Library File.

50.7.2.14 **Subsidiary Model Reference Record.** This optional record is used to designate another model within the model libraries as a subcomponent of this model.

50.7.2.15 **Point Light String Record.** This optional record is used to define each point light string within a model. It can be used to represent a single light by indicating that the number of lights is one. Point lights are light emitting objects represented spatially by a single coordinate within a model (e.g., a headlight on an automobile). They contain several attributes necessary for describing a light emitter such as the light lobe parameters, cycle rate, light type, and intensity. Point light strings are a sequence of discrete but logically connected light emitters (e.g., runway lights).

50.7.2.16 **Point Light String FACS Record.** This optional record contains a FACS (Feature Attribute Coding Standard) value associated with a point light string as a whole.

50.7.2.17 **Model FACS Record.** This optional record contains a FACS (Feature Attribute Coding Standard) value associated with a model as a whole (as opposed to a polygon within the model).

50.7.2.18 **Face-Based Texture Reference Record.** This optional record is used to define one method of placing a texture pattern on a polygon. It associates a model polygon with a texture in the Model Texture Library. The data contained in this record defines the transformation required to place a texture pattern on a polygon. All referenced textures will be Stage 3 specific model or generic textures.

50.7.2.19 **Vertex-to-Vertex Texture Reference Record.** This optional record is used to define another method of placing a texture pattern on a polygon. It associates the polygon with a texture in the Model Texture Library. This entails the mapping of texture pattern vertices to polygon vertices. There will be one texture pattern vertex defined for each polygon vertex. The first texture pattern vertex will map to the first polygon vertex. All referenced textures will be Stage 3 specific model or generic textures.

50.7.2.20 **Model-Based Texture Reference Record.** This optional record is used to define a method of placing a texture pattern as a single entity on a model. It associates the model with a texture in the Model Texture Library. This type of texturing can be conceptualized as the texture being "shrink-wrapped" onto the model. All referenced textures will be Stage 3 specific model or generic textures.

50.7.2.21 **Non-Mapped Texture Reference Record.** This record identifies a reference for a texture that is not directly mapped to the model. It associates a texture with a model.

50.7.2.22 **Checksum Record.** This record contains the checksum value for the 2-D Static Model Library File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the file. The checksum does not include those characters contained in the Checksum record itself.

50.8 **2-D Static Model Vertex (2DSMV) File.** The 2DSMV will consist of one or more 2DSMV Pseudo-files, corresponding to the number of models in the 2DSM. Each 2DSMV pseudo-file contains a list of the vertex coordinates used to define all LODs for a given model. These vertices are referenced from within the model definitions in the 2DSM, in order to define model polygons and vertex normals. The user will be able to associate a vertex pseudo-file with its model via a file naming convention. Each vertex pseudo-file name will take the form MDL2DSnnnnn.VTX, where 'nnnnn' corresponds to the unique model ID number. These pseudo-files will physically occur on the tape in the same sequence as their corresponding model definitions occur within the model library.

50.8.1 **2DSMV File Structure.** Self-explanatory.

50.8.2 **2DSM Record Structure.** Self-explanatory.

50.8.2.1 **Two-Dimensional Static Model Vertex Pseudo-Files.** These pseudo-files contain all coordinate vertices used to define models within the 2-D Static Model Library (2DSM) File.

50.8.2.1.1 **2DSMV Pseudo-File Record Order.** Self-explanatory.

50.8.2.1.2 **2DSMV Pseudo-File Field Structure.** Self-explanatory.

50.8.2.1.2.1 **2DSMV Identifier Record.** Self-explanatory.

50.8.2.1.2.2 **File Name Record.** Self-explanatory.

50.8.2.1.2.3 **Vertex Record.** Each Vertex record contains a coordinate used to define a model vertex within the 2DSM. The value in the Vertex List Position field is used as an index by a model referencing a vertex. The internal format of the Coordinate field will vary depending on the coordinate system selected for a particular GTDB. (Valid formats are defined within the Representation Dictionary in Appendix A to the DBDD.)

50.8.2.1.2.4 **Pseudo-EOF Record.** Self-explanatory.

50.8.2.1.2.5 **Checksum Record.** This record contains the checksum value for the Two-Dimensional Static Model Vertex Pseudo-File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the pseudo-file. The checksum does not include those characters contained in the Checksum record itself.

50.9 **3-D Static Model (3DSM) Library File.** The 3-D Static Model Library File is optional in a GTDB. When present, it contains all the 3-D static models referenced within the GTDB or explicitly requested during GTDB generation. The geometry of each model is represented as a set of surface polygons in 3-D space. The perimeter of each polygon is described by a set of coordinates or vertices. Each set of coordinates is identified by its position in a list of values stored in a separate 3-D Static Model Vertex File, allowing each coordinate set to be used repeatedly. By convention, all GTDB model polygons will be closed implicitly rather than explicitly; i.e., the first vertex of a polygon will not be explicitly listed again as the last vertex. Each surface or polygon may have descriptive and rendering attributes associated with it. A wide range of fields are available to describe attributes specific to radar, visual, and infrared simulation, as well as general attributes applicable to all three. Each polygon may reference any number of photo texture maps from the Model Photo Texture Library. The model library structure also supports composite models in which one model references another as a component. The 3DSM supports storage of models for any given object at multiple levels of detail. This will permit selection of models at a level of complexity that balances the desire for realism and the resolution of the simulated sensor with the processing capacity of the image generator and the overall complexity of the scene. The 3DSM includes a provision for identification of separation planes in three-dimensional space. These planes may be built between different objects within the model or they may be polygons which describe the model and, due to their position within it, also act as separating planes. It will also be possible to store 3-D models without separation planes.

50.9.1 **3DSM Record Order.** Self-explanatory.

50.9.2 **3DSM Field Structure.** Self-explanatory.

50.9.2.1 **3DSM Identifier Record.** Self-explanatory.

50.9.2.2 **3DSM File Name Record.** Self-explanatory.

50.9.2.3 **3DSM Header Record.** This mandatory record contains control information describing this file.

50.9.2.4 **Model Header Record.** This record identifies a model group, which can consist of one or more actual model geometries representing a given object at varying LODs.

50.9.2.5 **LOD Header Record.** This record describes a particular LOD version of a model. (Fields designated "Always Zero" are not applicable to 3-D static models. They have been included for consistency of format across all model libraries.)

50.9.2.6 **LOD Texture Reference Pointer Record.** This record provides pointers to the texture references for a model LOD.

50.9.2.7 **Component Header Record.** This record describes a particular component within a model LOD.

50.9.2.8 **Component Texture Reference Pointer Record.** Self-explanatory.

50.9.2.9 **Model Polygon Record.** This record describes the attributes of a polygon within a particular model. The last polygon record will always define the model "footprint" as referenced by the Base Polygon ID field in the parent Model record. This polygon is not meant to be displayed during a simulation but is supplied to aid the user in placing additional instances of the model. In addition to a unique Polygon ID, each polygon may have a Cluster ID which associates the polygon with a group of polygons which have been logically separated from the rest of the model by a separation plane. The Cluster ID also identifies a polygon's position relative to any of the separation planes defined by the Separation Plane Records associated with the model. See paragraph 50.9.2.20 for an explanation of the relationship between clusters and separation planes.

50.9.2.10 **Model Microdescriptor Record.** This optional record contains a microdescriptor associated with a model polygon. The types of microdescriptors supported by Project 2851 are listed in Section 5.16 of this standard.

50.9.2.11 **Vertex Pointer Record.** This record is used to associate a model polygon with a Vertex record within the Three-Dimensional Static Model Vertex File. There will be three or more of these records defining the geometry of each model polygon. By convention, a model polygon will be closed implicitly rather than explicitly; i.e., the first vertex of a polygon will not be explicitly referenced again as the last vertex. The Vertex List Position and Correlation Priority fields apply to the coordinate defining a polygon vertex. The Normal List Position field points to a separate 3DSMV coordinate defining a vertex normal vector. This field will be populated with meaningful data only when vertex normals have been requested.

50.9.2.12 **Polygon FACS Record.** This optional record contains a FACS (Feature Attribute Coding Standard) value associated with a model polygon.

50.9.2.13 **Polygon Texture Reference Record.** This optional record is used to associate a model polygon with a Photo Texture Image Header record within the Model Photo Texture Library File.

50.9.2.14 **Subsidiary Model Reference Record.** This optional record is used to designate another model within the model libraries as a subcomponent of this model.

50.9.2.15 **Point Light String Record.** Self-explanatory.

50.9.2.16 **Point Light String FACS Record.** Self-explanatory.

50.9.2.17 **Model FACS Record.** This optional record contains a FACS (Feature Attribute Coding Standard) value associated with a model as a whole (as opposed to a polygon within the model).

50.9.2.18 **Face-Based Texture Reference Record.** Self-explanatory.

50.9.2.19 **Vertex-to-Vertex Texture Reference Record.** Self-explanatory.

50.9.2.20 **Model-Based Texture Reference Record.** Self-explanatory.

50.9.2.21 **Non-Mapped Texture Reference Record.** Self-explanatory.

50.9.2.22 **Separation Plane Record.** This optional record is used to define a separation plane within a model. Separation planes are used to divide a 3-dimensional model into distinct clusters of model polygons, which provide a basis for efficient display-priority resolution when the model is rendered on a graphics device.

50.9.2.22.1 A Separation Plane Number indicates the position of a separating plane within a binary separating plane (BSP) tree. An example of a BSP tree is illustrated below. At every level of the tree, the left child of a parent tree node represents the "true" (i.e., visible) half-space, or side, of the plane, while the right child represents the false side of the plane.

50.9.2.22.2 In the example, the root node (SP 1) of the BSP tree by itself divides the entire model into two half-spaces or clusters. The root node is shown having a left child and a right child. The left child divides the true cluster of the root node plane into two more clusters. The right child plane does the same to the false cluster of the root node plane. Finally, the right child plane has a left child of its own, dividing that plane's true cluster into two more clusters.

```
        SP 1
       /    \
      /      \
  SP 2       SP 3
             /
            /
        SP 6
```

50.9.2.22.3 As mentioned above, each node, or plane, has an identifying separation plane number which represents its position in the BSP tree. This number is determined by counting from top to bottom within the tree, from the left-most node to the right-most node at each level, as if the tree were complete (i.e., with all levels filled). This explains why the lowest node in the example is numbered 6 and not 4.

50.9.2.22.4 Note that the order of creation of the planes may be partially independent of the position of the planes in the tree, and hence of the separation plane numbers. Of course, the very first separation plane created for a model would have to be the root node and be assigned plane number 1. At lower levels, however, the nodes could be defined in any order, so long as any given node's parent has been previously defined. Within a model, the separation plane records will be physically ordered not by separation plane number but by the order in which the planes were created.

50.9.2.22.5 This order is important when determining Cluster IDs associated with every Model Polygon Record. The Cluster ID may be used to determine a polygon's position relative to the separation planes. A polygon can be on the true side of a plane, the false side, or in a "don't care" position. (This requires that polygons intersecting the plane be cut to lie entirely on one side or the other.) The "don't care" case occurs when a polygon has already been eliminated from the area of concern of a separating plane by a previously placed plane. A Cluster ID is a string of binary digits in which, if a polygon lies to the true side of the nth plane in the ordered list of planes, then the nth low-order bit is set to '1'; otherwise, the bit is set to '0'.

50.9.2.22.6 Continuing our example BSP tree, and assuming that the separating planes were defined in the order 1,2,3,6, the assigned Cluster IDs are shown below.

```
               SP 1
              /    \
             /      \
        SP 2          SP 3
        / \            / \
       /   \          /   \
   0011   0001     SP 6   0000
                   / \
                  /   \
               1100   0100
```

50.9.2.22.7 Note that in this example the list is ordered such that plane numbers are always increasing. As previously noted, this need not always be the case. Consider adding another plane at this point which would be the left child of plane number 2. This newest plane would have a plane number of 4 based on its position in the BSP tree, but it would be treated as the fifth plane in the sequence of separation planes for purposes of generating cluster IDs. The resulting BSP tree and Cluster IDs would be as follows.

```
                  SP 1
                 /    \
                /      \
            SP 2        SP 3
             / \         / \
            /   \       /   \
        SP 4  00001  SP 6  00000
         / \          / \
        /   \        /   \
    10011  00011  01100  00100
```

50.9.2.22.8 For further details on Project 2851 model separating planes, consult the modeling software documentation.

50.9.2.23 **Checksum Record.** This record contains the checksum value for the 3-D Static Model Library File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the file. The checksum does not include those characters contained in the Checksum record itself.

50.10 **Three-Dimensional Static Model Vertex (3DSMV) File.** This file consists of a series of pseudo-files containing all coordinate vertices used to define models within the Three-Dimensional Static Model Library File. Each 3DSMV pseudo-file contains a list of the vertex coordinates used to define all LODs for a given model. These vertices are referenced from within the model definitions in the 3DSM, in order to define model polygons, vertex normals, and separation planes. The user will be able to associate a vertex pseudo-file with its model via a file naming convention. Each vertex pseudo-file name will take the form MDL3DSnnnnn.VTX, where 'nnnnn' corresponds to the unique model ID number. These pseudo-files will physically occur on the tape in the same sequence as their corresponding model definitions occur within the model library.

50.10.1 **3DSMV File Structure.** Self-explanatory.

50.10.2 **3DSM Record Structure.** Self-explanatory.

50.10.2.1 **3DSMV Pseudo-Files.** These pseudo-files contain all coordinate vertices used to define models within the Three-Dimensional Static Model (3DSM) Library File.

50.10.2.1.1 **3DSMV Pseudo-File Record Order.** Self-explanatory.

50.10.2.1.2 **3DSMV Pseudo-File Field Structure.** Self-explanatory.

50.10.2.1.2.1 **3DSMV Identifier Record.** Self-explanatory.

50.10.2.1.2.2 **File Name Record.** Self-explanatory.

50.10.2.1.2.3 **Vertex Record.** Each Vertex record contains a coordinate used to define a model vertex or a vertex normal within the 3DSM. The value in the Vertex List Position field is used as an index by a model referencing a vertex. The internal format of the Coordinate field will vary depending on the coordinate system selected for a particular GTDB. (Valid formats are defined within the Representation Dictionary in Appendix A of this standard.)

50.10.2.1.2.4 **Pseudo-EOF Record.** Self-explanatory.

50.10.2.1.2.5 **Checksum Record.** This record contains the checksum value for the Three-Dimensional Static Model Vertex Pseudo-File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the pseudo-file. The checksum does not include those characters contained in the Checksum record itself.

50.11 **3-D Dynamic Model (3DDM) Library File.** The 3-D Dynamic Model Library File is optional in a GTDB. When present, it contains all the 3-D dynamic models referenced within the GTDB or explicitly requested during GTDB generation. The geometry of each model is represented as a set of surface polygons in 3-D space. The perimeter of each polygon is described by a set of coordinates or vertices. Each set of coordinates is identified by its position in a list of values stored in a separate 3-D Dynamic Model Vertex File, allowing each coordinate set to be used repeatedly. By convention, all GTDB model polygons will be closed implicitly rather than explicitly; i.e., the first vertex of a polygon will not be explicitly listed again as the last vertex. Each surface or polygon may have descriptive and rendering attributes associated with it. A wide range of fields are available to describe attributes specific to radar, visual, and infrared simulation, as well as general attributes applicable to all three. Each polygon may reference any number of photo texture maps from the Model Photo Texture Library. The model library structure also supports composite models in which one model references another as a component. The 3DDM supports storage of models for any given object at multiple LODs. This will permit selection of models at a level of complexity that balances the desire for realism and the resolution of the simulated sensor with the processing capacity of the image generator and the overall complexity of the scene. From a data structure standpoint, three-dimensional static models and three-dimensional dynamic models in the GTDB are identical, except that the dynamic models identify coordinates which may be used for collision detection tests.

50.11.1 **3DDM Record Order.** Self-explanatory.

50.11.2 **3DDM Field Structure.** Self-explanatory.

50.11.2.1 **3DDM Identifier Record.** Self-explanatory.

50.11.2.2 **File Name Record.** Self-explanatory.

50.11.2.3 **3DDM Header Record.** This mandatory record contains control information describing this file.

50.11.2.4 **Model Header Record.** This record identifies a model group, which can consist of one or more actual model geometries representing a given object at varying LODs.

50.11.2.5 **LOD Header Record.** This record describes a particular LOD version of a model.

50.11.2.6 **LOD Texture Reference Pointer Record.** This record provides pointers to the texture references for a model LOD.

50.11.2.7 **Component Header Record.** This record describes a particular component within a model LOD.

50.11.2.8 **Component Texture Reference Pointer Record.** Self-explanatory.

50.11.2.9 **Model Polygon Record.** This record describes the attributes of a polygon within a particular model. The last polygon record will always define the model "footprint" as referenced by the Base Polygon ID field in the parent Model record. This polygon is not meant to be displayed during a simulation but is supplied to aid the user in placing additional instances of the model. In addition to a unique Polygon ID, each polygon may have a Cluster ID which associates the polygon with a group of polygons which have been logically separated from the rest of the model by a separation plane. The Cluster ID also identifies a polygon's position relative to any of the separation planes defined by the Separation Plane Records associated with the model. See paragraph 50.9.2.20 for an explanation of the relationship between clusters and separation planes.

50.11.2.10 **Model Microdescriptor Record.** This optional record contains a microdescriptor associated with a model polygon. The types of microdescriptors supported by Project 2851 are listed in Section 5.16 of this standard.

50.11.2.11 **Vertex Pointer Record.** This record is used to associate a model polygon with a Vertex record within the Three-Dimensional Dynamic Model Vertex (3DDMV) File. There will be three or more of these records defining the geometry of each model polygon. By convention, a model polygon will be closed implicitly rather than explicitly; i.e., the first vertex of a polygon will not be explicitly referenced again as the last vertex. The Vertex List Position and Correlation Priority fields apply to the coordinate defining a polygon vertex. The Normal List Position field points to a separate 3DDMV coordinate defining a vertex normal vector. This field will be populated with meaningful data only when vertex normals have been requested.

50.11.2.12 **Polygon FACS Record.** This optional record contains a FACS (Feature Attribute Coding Standard) value associated with a model polygon.

50.11.2.13 **Polygon Texture Reference Record.** This optional record is used to associate a model polygon with a Photo Texture Image Header record within the Model Photo Texture Library File.

50.11.2.14 **Subsidiary Model Reference Record.** This optional record is used to designate another model within the model libraries as a subcomponent of this model.

50.11.2.15 **Point Light String Record.** Self-explanatory.

50.11.2.16 **Point Light String FACS Record.** Self-explanatory.

50.11.2.17 **Model FACS Record.** This optional record contains a FACS (Feature Attribute Coding Standard) value associated with a model as a whole (as opposed to a polygon within the model).

50.11.2.18 **Face-Based Texture Reference Record.** Self-explanatory.

50.11.2.19 **Vertex-to-Vertex Texture Reference Record.** Self-explanatory.

50.11.2.20 **Model-Based Texture Reference Record.** Self-explanatory.

50.11.2.21 **Non-Mapped Texture Reference Record.** Self-explanatory.

50.11.2.22 **Separation Plane Record.** This optional record is used to define a separation plane within a model. (See section 50.9.2.20 of this standard for an extended discussion of the separation plane record.)

50.11.2.23 **Collision Test Point Record.** This optional record is used to designate a Vertex record within the Three-Dimensional Dynamic Model Vertex (3DDMV) File as a collision test point for a model.

50.11.2.24 **Checksum Record.** This record contains the checksum value for the 3-D Dynamic Model Library File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the file. The checksum does not include those characters contained in the Checksum record itself.

50.12 **Three-Dimensional Dynamic Model Vertex (3DDMV) File.** This file consists of a series of pseudo-files containing all coordinate vertices used to define models within the Three-Dimensional Dynamic Model (3DDM) Library File. The 3DDMV will consist of one or more 3DDMV Pseudo-files, corresponding to the number of models in the 3DDM. Each 3DDMV pseudo-file contains a list of the vertex coordinates used to define all LODs for a given model. These vertices are referenced from within the model definitions in the 3DDM, in order to define model polygons, vertex normals, separation planes, and collision test points. The user will be able to associate a vertex pseudo-file with its model via a file naming convention. Each vertex pseudo-file name will take the form MDL3DDnnnnn.VTX, where 'nnnnn' corresponds to the unique model ID number. These pseudo-files will physically occur on the tape in the same sequence as their corresponding model definitions occur within the model library.

50.12.1 **3DDMV File Structure.** Self-explanatory.

50.12.2 **3DDMV Record Structure.** Self-explanatory.

50.12.2.1 **3-D Dynamic Model Vertex Pseudo-Files.** These pseudo-files contain all coordinate vertices used to define models within the Three-Dimensional Dynamic Model (3DDM) Library File.

50.12.2.1.1 **3DDMV Pseudo-File Record Order.** Self-explanatory.

50.12.2.1.2 **3DDMV Pseudo-File Field Structure.** Self-explanatory.

50.12.2.1.2.1 **3DDMV Identifier Record.** Self-explanatory.

50.12.2.1.2.2 **File Name Record.** Self-explanatory.

50.12.2.1.2.3 **Vertex Record.** Each Vertex record contains a coordinate used to define a model vertex, a vertex normal, or a collision test point, within the 3DDM. The value in the Vertex List Position field is used as an index by a model referencing a vertex. The internal format of the Coordinate field will vary depending on the coordinate system selected for a particular GTDB. (Valid formats are defined within the Representation Dictionary in Appendix A to this standard.)

50.12.2.1.2.4 **Pseudo-EOF Record.** Self-explanatory.

50.12.2.1.2.5 **Checksum Record.** This record contains the checksum value for the Three-Dimensional Dynamic Model Vertex Pseudo-File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the pseudo-file. The checksum does not include those characters contained in the Checksum record itself.

50.13 **Simulator Level of Detail Area Blocks File (SLAB).** This mandatory file contains all applicable terrain and culture area blocks defined for a given Simulator Level of Detail (SLOD). There may be one or more Simulator Level of Detail Area Blocks Files in a valid GTDB. The data within each SLAB are organized by area block, and within each area block as a collection of pseudo-files, each containing a particular type of data for that area block. There are nine varieties of pseudo-files possible per area block. The exact configuration of pseudo-files for a given simulator level of detail or area block will depend on the transformation parameters used and on the availability of specific types of data in the Standard Simulator Data Base. Each of the pseudo-file types is described in the following subsections, in the sequence in which they would occur on a GTDB tape. Each SLAB file is structured as a collection of pseudo-files, separated by pseudo-EOF records. The pseudo-files are organized to describe each area block making up the simulator level of detail in sequence, with an Area Block Pseudo-EOF Record separating the area blocks.

50.13.1 **SLAB File Structure.** Self-explanatory.

50.13.2 **SLAB Pseudo-File Record Structure.** Self-explanatory.

50.13.2.1 **Vertex Area Block (VAB) Pseudo-File.** The VAB contains a list of the vertex coordinates used to define all culture features within a given area block. In addition, when polygonized (as opposed to gridded) terrain has been requested, the vertex area block will also contain the vertex coordinates used to define the terrain polygons and terrain vertex normals. The vertices in the vertex area block are referenced from within the AFAB, LFAB, PFAB, PLFAB, PLSFAB, TPAB, and MRAB pseudo-files, as defined below. There will always be at least four vertices defined for an area block, since there will always be at least one areal feature (the background feature) in an area block. Each vertex coordinate is expressed in 3-D space, having x, y, and z components in the context of the selected coordinate system. However, the z-component for culture vertices will be zero except in two user-specified circumstances: (a) when the culture has been fragmented on the underlying terrain polygons, or (b) if the coordinate system is geocentric. The vertex area block for the first area block within a simulator level of detail will be the first pseudo-file in the SLAB. The vertex area block file for succeeding area blocks (if any) will follow the MRAB file of the preceding area block. The vertex records in this file are referenced by the various feature and terrain pseudo-files defined for a particular area block.

50.13.2.1.1 **VAB Pseudo-File Record Order.** Self-explanatory.

50.13.2.1.2 **VAB Pseudo-File Field Structure.** Self-explanatory.

50.13.2.1.2.1 **VAB Identifier Record.** Self-explanatory.

50.13.2.1.2.2 **File Name Record.** Self-explanatory.

50.13.2.1.2.3 **Vertex Area Block Header Record.** This mandatory record contains a count of the number of vertices in the area block and the Vertex ID of the last vertex.

50.13.2.1.2.4 **Vertex Record.** Each Vertex record contains a coordinate used to define a culture or terrain vertex or vertex normal within the area block. The value in the Vertex List Position field is used as an index by feature and terrain files referencing a vertex. The internal format of the Coordinate field will vary depending on the coordinate system selected for a particular GTDB. (Valid formats are defined within the Representation Dictionary in Appendix A to this standard.)

50.13.2.1.2.5 **Pseudo-EOF Record.** Self-explanatory.

50.13.2.1.2.6 **Checksum Record.** This record contains the checksum value for the vertex area block Pseudo-File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the pseudo-file. The checksum does not include those characters contained in the Checksum record itself.

50.13.2.2 **Areal Feature Area Block (AFAB) Pseudo-File.** The Areal Feature Area Block Pseudo-File is mandatory for every area block in a GTDB. The AFAB contains all areal features included within an area block. This is a required file, as there will always be at least one areal feature (the background feature) in an area block. The AFAB will follow the vertex area block for the area block. Generally, there will be many areal features in a block. Each feature will have a unique number associated with it. An option available to the GTDB user will cause areal features to be replaced by model references where suitable models have been defined within the SSDB. When model substitution occurs, the areal feature will be eliminated from the AFAB, and a model reference will be inserted in the MRAB. Another transformation option available to the user will cause areal features to be fragmented along the underlying terrain polygon boundaries. When such fragmentation occurs, each fragment becomes an individual areal feature with a unique feature number. At the same time, a fragmentation flag will be set, and a common "superfeature number" will be assigned to the fragments, so that the complete feature can be reconstructed if desired. The geometry of each feature will be described by vertex pointers describing its shape and location within the area block. The "footprint" of each areal feature will be defined by three or more vertices in the coordinate space of the area block. The vertices will be listed in counterclockwise order as viewed from above. By convention, all areal features will be closed implicitly rather than explicitly; i.e., the first vertex of the feature will not be explicitly listed again as the last vertex. Individual vertices are reusable and stored as coordinate triplets in the Vertex Area Block Pseudo-File, where they are referenced by their list position. This sharing of common vertices helps ensure spatial consistency of terrain and culture data elements, and also saves storage space. Each feature may be described by numerous attributes, some of which have general applicability, and others of which apply to specific sensors. As applicable, a feature may have a wide variety of FACS attributes and microdescriptors. FACS attributes are supplementary attributes not contained among the "core" descriptors. Microdescriptors are a specialized class of feature attribution mechanisms which support specification of complex sub-detail. An areal feature may also have any number of texture codes associated with it to indicate which areal photo-texture maps from the areal photo-texture library will be applied to it in real-time simulation. Each texture code is associated with a pattern origin which indicates where the mapping starts.

50.13.2.2.1 **AFAB Pseudo-File Record Order.** Self-explanatory.

50.13.2.2.2 **AFAB Pseudo-File Field Structure.** Self-explanatory.

50.13.2.2.2.1 **AFAB Identifier Record.** Self-explanatory.

50.13.2.2.2.2 **File Name Record.** Self-explanatory.

50.13.2.2.2.3 **Feature Area Block Header Record.** This record contains control information describing the file.

50.13.2.2.2.4 **Face-Based Texture Reference Record.** This optional record is used to define a method of placing a texture pattern on an areal feature polygon. The record associates an areal feature polygon with a texture in the Areal Texture Library. The data contained in this record defines the transformation required to place a texture on the polygon. This record supports generic texture in Stage 5 where the texture is mapped to polygons. Stage 5 is simply Stage 3 or 4 with polygon mapping information.

50.13.2.2.2.5 **Global-Based Texture Reference Record.** This optional record is used to define a method of placing a texture pattern as a single entity on a homogeneous cultural area made up of one or more culture features. The record associates these homogeneous areal features with a texture in the Areal Texture Library. This type of texturing can be conceptualized as the texture being "shrink-wrapped" onto the features. This record supports generic texture in Stage 5 where the texture is mapped to polygons. Stage 5 is simply Stage 3 or 4 with polygon mapping information.

50.13.2.2.2.6 **Areal Feature Record.** This record contains control fields and attributes describing a particular areal feature.

50.13.2.2.2.7 **Microdescriptor Record.** This optional record contains a microdescriptor associated with a feature. The types of microdescriptors supported by the GTDB are listed in Section 5.16 of this standard.

50.13.2.2.2.8 **FACS Record.** This optional record contains a FACS (Feature Attribute Coding Standard) value associated with a feature.

50.13.2.2.2.9 **Vertex Pointer Record.** This record is used to associate a feature with a Vertex record within the Vertex Area Block Pseudo-File. There will be three or more of these records defining the geometry of each areal feature. By convention, an areal feature will be closed implicitly rather than explicitly; i.e., the first vertex of a feature will not be explicitly referenced again as the last vertex.

50.13.2.2.2.10 **Non-Mapped Texture Reference Record.** This optional record is used to associate a feature with a generic texture within the Areal Texture Library. This record supports generic texture in Stages 3 and 4 where the texture is not mapped to any polygons. (Generic texture does not exist at Stages 1 and 2.)

50.13.2.2.2.11 **Mapped Texture Reference Pointer Record.** This optional record is used to point to a texture reference for an areal feature. This record supports generic texture in Stage 5 where the texture is mapped to polygons. (Generic texture does not exist at Stages 1 and 2.)

50.13.2.2.2.12 **Pseudo-EOF Record.** Self-explanatory.

50.13.2.2.2.13 **Checksum Record.** This record contains the checksum value for the Areal Feature Area Block Pseudo-File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the pseudo-file. The checksum does not include those characters contained in the Checksum record itself.

50.13.2.3 **Linear Feature Area Block (LFAB) Pseudo-File.** The Linear Feature Area Block Pseudo-File contains all linear features included within an area block. Options available to the GTDB user will cause linear features to be replaced by areal features and/or model references where alternate representations have been defined within the SSDB. When feature substitution occurs, the linear feature will be eliminated from the LFAB, and an alternate will be inserted in the AFAB or MRAB. When present, the LFAB will follow the AFAB for the area block. A linear feature will be described by many of the same types of descriptive attributes as areal features. Unlike areals, however, each linear feature's spatial position will be described by a series of connected coordinate vertices which do not have to form a closed polygon. To save storage space, vertices are reusable and stored as coordinate triplets in the Vertex Area Block Pseudo-File, where they are referenced by their list position.

50.13.2.3.1 **LFAB Pseudo-File Record Order.** Self-explanatory.

50.13.2.3.2 **LFAB Pseudo-File Field Structure.** Self-explanatory.

50.13.2.3.2.1 **LFAB Identifier Record.** Self-explanatory.

50.13.2.3.2.2 **File Name Record.** Self-explanatory.

50.13.2.3.2.3 **Feature Area Block Header Record.** Self-explanatory.

50.13.2.3.2.4 **Linear Feature Record.** This record contains control fields and attributes describing a particular linear feature.

50.13.2.3.2.5 **Microdescriptor Record.** This optional record contains a microdescriptor associated with a feature. The types of microdescriptors supported by the GTDB are listed in Section 5.16 of this standard.

50.13.2.3.2.6 **FACS Record.** This optional record contains a FACS (Feature Attribute Coding Standard) value associated with a feature.

50.13.2.3.2.7 **Vertex Pointer Record.** This record is used to associate a feature with a Vertex record within the Vertex Area Block Pseudo-File. There will be two or more of these records defining the geometry of each linear feature.

50.13.2.3.2.8 **Non-Mapped Texture Reference Record.** This optional record is used to associate a feature with a Photo Texture Image Header record within the Areal Photo Texture Library File. This optional record is used to associate a feature with a generic texture within the Areal Texture Library. This record supports generic texture in Stages 3 and 4 where the texture is not mapped to any polygons. (Generic texture does not exist at Stages 1 and 2.) While textures cannot be mapped to linear features, they are provided here in order to aid the GTDB user who might expand a linear feature into an areal feature.

50.13.2.3.2.9 **Pseudo-EOF Record.** Self-explanatory.

50.13.2.3.2.10 **Checksum Record.** This record contains the checksum value for the Linear Feature Area Block Pseudo-File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the pseudo-file. The checksum does not include those characters contained in the Checksum record itself.

50.13.2.4 **Point Feature Area Block (PFAB) Pseudo-File.** The Point Feature Area Block Pseudo-File contains all point features included within an area block. An option available to the GTDB user will cause point features to be replaced by model references where suitable models have been defined within the SSDB. When model substitution occurs, the point feature will be eliminated from the PFAB, and a model reference will be inserted in the MRAB. When present, the PFAB will follow the LFAB for the area block. Point features will be described by many of the same types of descriptive attributes as areal and linear features. However, a point feature's spatial position will normally be specified by a single coordinate. To be consistent with a convention established by the Defense Mapping Agency (DMA), the PFAB will also be used to store features consisting of a sequence of physically discrete points (e.g., a string of transmission towers). To save storage space, vertices are reusable and stored as coordinate triplets in the Vertex Area Block Pseudo-File, where they are referenced by their list position.

50.13.2.4.1 **PFAB Pseudo-File Record Order.** Self-explanatory.

50.13.2.4.2 **PFAB Pseudo-File Field Structure.** Self-explanatory.

50.13.2.4.2.1 **PFAB Identifier Record.** Self-explanatory.

50.13.2.4.2.2 **File Name Record.** Self-explanatory.

50.13.2.4.2.3 **Feature Area Block Header Record.** This mandatory record contains control information describing the file.

50.13.2.4.2.4 **Point Feature Record.** This record contains control fields and attributes describing a particular point feature.

50.13.2.4.2.5 **Microdescriptor Record.** This optional record contains a microdescriptor associated with a feature. The types of microdescriptors supported by the GTDB are listed in Section 5.16 of this standard.

50.13.2.4.2.6 **FACS Record.** This optional record contains a FACS (Feature Attribute Coding Standard) value associated with a feature.

50.13.2.4.2.7 **Vertex Pointer Record.** This record is used to associate a feature with a Vertex record within the Vertex Area Block Pseudo-File. There will be one or more of these records defining the geometry of each point feature. (Although a true point feature can be defined by a single coordinate, by convention a point feature may consist of a collection of related but non-connected points, each member of which will be designated as a vertex.)

50.13.2.4.2.8 **Non-Mapped Texture Reference Record.** This optional record is used to associate a feature with a generic texture within the Areal Texture Library. This record supports generic texture in Stages 3 and 4 where the texture is not mapped to any polygons. (Generic texture does not exist at Stages 1 and 2.) While textures cannot be mapped to point features, they are provided here in order to aid the GTDB user who might expand a point feature into an areal feature.

50.13.2.4.2.9 **Pseudo-EOF Record.** Self-explanatory.

50.13.2.4.2.10 **Checksum Record.** This record contains the checksum value for the Point Feature Area Block Pseudo-File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the pseudo-file. The checksum does not include those characters contained in the Checksum record itself.

50.13.2.5 **Point Light Feature Area Block (PLFAB) Pseudo-File.** The Point Light Feature Area Block Pseudo-File contains all point light features included within an area block. An option available to the GTDB user will cause point light features to be replaced by model references where suitable models have been defined within the SSDB. When model substitution occurs, the point light feature will be eliminated from the PLFAB, and a model reference will be inserted in the MRAB. When present, the PLFAB will follow the PFAB for the area block. Point light features will be described by many of the same types of descriptive attributes as regular point features, along with additional attributes describing the light emitter. To save storage space, vertices are reusable and stored as coordinate triplets in the Vertex Area Block Pseudo-File, where they are referenced by their list position.

50.13.2.5.1 **PLFAB Pseudo-File Record Order.** Self-explanatory.

50.13.2.5.2 **PLFAB Pseudo-File Field Structure.** Self-explanatory.

50.13.2.5.2.1 **PLFAB Identifier Record.** Self-explanatory.

50.13.2.5.2.2 **File Name Record.** Self-explanatory.

50.13.2.5.2.3 **Feature Area Block Header Record.** Self-explanatory.

50.13.2.5.2.4 **Point Light Feature Record.** Self-explanatory.

50.13.2.5.2.5 **Microdescriptor Record.** This optional record contains a microdescriptor associated with a feature. The types of microdescriptors supported by the GTDB are listed in Section 5.16 of this standard.

50.13.2.5.2.6 **FACS Record.** This optional record contains a FACS (Feature Attribute Coding Standard) value associated with a feature.

50.13.2.5.2.7 **Vertex Pointer Record.** This record is used to associate a feature with a Vertex record within the Vertex Area Block Pseudo-File.

50.13.2.5.2.8 **Pseudo-EOF Record.** Self-explanatory.

50.13.2.5.2.9 **Checksum Record.** This record contains the checksum value for the Point Light Feature Area Block Pseudo-File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the pseudo-file. The checksum does not include those characters contained in the Checksum record itself.

50.13.2.6 **Point Light String Feature Area Block (PLSFAB) Pseudo-File.** The Point Light String Feature Area Block Pseudo-File contains all point light string features included within an area block. An option available to the GTDB user will cause point light string features to be fragmented into a series of individual point light features. When fragmentation occurs, the PLSFAB will be eliminated from the GTDB, and the new point light features will be inserted in the PLFAB. When present, the PLSFAB will follow the PLFAB for the area block. Point light string features will be described by many of the same types of descriptive attributes as point light features, along with additional attributes describing the shape and orientation of the string of lights. To save storage space, vertices are reusable and stored as coordinate triplets in the Vertex Area Block Pseudo-File, where they are referenced by their list position.

50.13.2.6.1 **PLSFAB Pseudo-File Record Order.** Self-explanatory.

50.13.2.6.2 **PLSFAB Pseudo-File Field Structure.** Self-explanatory.

50.13.2.6.2.1 **PLSFAB Identifier Record.** Self-explanatory.

50.13.2.6.2.2 **File Name Record.** Self-explanatory.

50.13.2.6.2.3 **Feature Area Block Header Record.** This mandatory record contains control information describing the file.

50.13.2.6.2.4 **Point Light String Feature Record.** This record contains control fields and attributes describing a particular point light string feature.

50.13.2.6.2.5 **Microdescriptor Record.** This optional record contains a microdescriptor associated with a feature. The types of microdescriptors supported by the GTDB are listed in Section 5.16 of this standard.

50.13.2.6.2.6 **FACS Record.** This optional record contains a FACS (Feature Attribute Coding Standard) value associated with a feature.

50.13.2.6.2.7 **Vertex Pointer Record.** This record is used to associate a feature with a Vertex record within the Vertex Area Block Pseudo-File. There will be one or more of these records defining the location of each point light within the point light string feature. The first record will always define the origin of the point light string. Any subsequent records will define succeeding points in the string. Vertex Pointer records will not be used to define points other than the origin when the Light String Shape Field within the parent Point Light String Feature record indicates that the pattern of lights in the string is a regular straight line. In that case, the Number of Lights and Light Delta fields should be used to determine the position of succeeding points in the string.

50.13.2.6.2.8 **Pseudo-EOF Record.** Self-explanatory.

50.13.2.6.2.9 **Checksum Record.** This record contains the checksum value for the Point Light String Feature Area Block Pseudo-File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the pseudo-file. The checksum does not include those characters contained in the Checksum record itself.

50.13.2.7 **Terrain Polygon Area Block (TPAB) Pseudo-File.** The TPAB contains all terrain polygons included within an area block. The choice of terrain polygons versus a terrain grid is up to the user. (Optionally, the user may request a GTDB containing both polygonized and gridded terrain, or may choose to omit terrain altogether.) This is a required file when polygonized terrain has been requested, but it will be omitted when only gridded terrain has been requested. When present, the TPAB will follow the PLSFAB for the area block. The TPAB will describe a continuous network of polygons representing the surface of the terrain within an area block. The type and density of terrain polygons is not limited by the data base design but will be specifiable by the user only within the allowable range of Project 2851 transformation parameters. (The common data base transformation program limits the polygon shape to triangles. Details of the terrain polygonization algorithm supported by Project 2851 are described in the software detailed design document for the common data base transformation program.) Each terrain polygon will be defined by its vertices in three-dimensional space. The vertices will be listed in counterclockwise order as viewed from above. By convention, all polygons will be closed implicitly rather than explicitly; i.e., the first vertex of the polygon will not be explicitly listed again as the last vertex. Vertices are reusable and stored as coordinate triplets in the Vertex Area Block File, where they may be referenced by their list position. Each terrain polygon record will have a surface normal vector included within it. At the requestor's option, a vertex normal vector will be calculated and associated with each terrain polygon vertex. These vertex normal vectors will also be stored as coordinate triplets in the Vertex Area Block File and referenced from the TPAB. (The vertex normal references will actually always be present but will be set to the value '0,0,0' when not specifically requested via the transformation parameters.) When culture fragmentation is requested, each terrain polygon will have associated with it a list of culture features and model references which lie upon it. These will be pointers to specific records in the various culture area block pseudo-files (AFAB, LFAB, PFAB, PLFAB, PLSFAB, and MRAB).

50.13.2.7.1 **TPAB Pseudo-File Record Order.** Self-explanatory.

50.13.2.7.2 **TPAB Pseudo-File Field Structure.** Self-explanatory.

50.13.2.7.2.1 **TPAB Identifier Record.** Self-explanatory.

50.13.2.7.2.2 **File Name Record.** Self-explanatory.

50.13.2.7.2.3 **TPAB Header Record.** This mandatory record contains control information describing the TPAB. The field structure of this record is given below. (Fields identified "Always Zero" are applicable to gridded rather than polygonized terrain. They have been included for consistency of format between the TPAB and TGAB area block headers. The Latitude Interval and Longitude Interval fields are used to describe the intervals in the standard simulator data base grid used to generate the TPAB.)

50.13.2.7.2.4 **Terrain Polygon Record.** Each Terrain Polygon record describes a single terrain polygon.

50.13.2.7.2.5 **Vertex List Pointer Record.** This record is used to associate a terrain polygon with a Vertex record within the Vertex Area Block (VAB) Pseudo-File. There will be three of these records defining the geometry of each terrain polygon. (The common data base transformation program limits the polygon shape to triangles; however, the Project 2851 data structure supports more complex polygons.) By convention, a terrain polygon will be closed implicitly rather than explicitly; i.e., the first vertex of a polygon will not be explicitly referenced again as the last vertex. The vertex list pointer records for a terrain polygon will be sequenced in counterclockwise order as viewed from above. The Vertex List Position and Correlation Priority fields apply to the coordinate defining a polygon vertex. The Normal List Position field points to a separate vertex area block coordinate defining a vertex normal vector. This field will be populated with meaningful data only when vertex normals have been requested.

50.13.2.7.2.6 **Culture Reference Record.** This record is used to associate a terrain polygon with a feature record within the AFAB, LFAB, PFAB, PLFAB, or PLSFAB pseudo-files, or with a model reference record in the MRAB pseudo-file. Each record indicates that a particular feature or model lies upon a particular terrain polygon.

50.13.2.7.2.7 **Vertex-to-Vertex Texture Reference Record.** This optional record is used to define the placement of a texture pattern on a polygon. It associates the polygon with a texture in the Areal Texture Library. This entails the mapping of texture pattern vertices to polygon vertices. This record supports specific areal texture and SMC/FDC texture for Stage 5. Stage 5 is simply Stage 3 or 4 with polygon mapping information.

50.13.2.7.2.8 **Pseudo-EOF Record.** Self-explanatory.

50.13.2.7.2.9 **Checksum Record.** This record contains the checksum value for the Terrain Polygon Area Block Pseudo-File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the pseudo-file. The checksum does not include those characters contained in the Checksum record itself.

50.13.2.8 **Terrain Grid Area Block (TGAB) Pseudo-File.** The TGAB contains all terrain grid posts included within an area block. (Optionally, the user may request a GTDB containing both polygonized and gridded terrain, or may choose to omit terrain altogether.) This is a required file when gridded terrain has been requested, but it will be omitted when only polygonized terrain has been requested. When present, the TGAB will follow the TPAB for the area block. The TGAB will represent the terrain within the area block as a systematically spaced grid of elevation values. The resolution (spacing) of the terrain posts will be specifiable by the user from among the available terrain LODs in the SSDB. The standard simulator data base has been designed to store terrain data, if available, at nominal post spacings of 100m, 30m, 10m, and 1m along x and y. The common data base transformation program will attempt to retrieve terrain data from the standard simulator data base LOD specified by the user for an area block. If such data do not exist, the common data base transformation program will select the best (highest resolution) lower LOD terrain which does exist in the SSDB. (A GTDB will not be created if standard simulator data base terrain data does not exist at all within the area block.) Where the standard simulator data base has only partial coverage at a requested resolution within an area block, such data will be used, with the remainder of the area block derived from lower resolution data. The TGAB will contain a complete grid at the user-specified post spacing, regardless of which standard simulator data base LODs are used as the source. The common data base transformation program will use linear interpolation to fill the grid if necessary. The TGAB is logically a standalone file within the GTDB, containing no pointers or references to other files.

50.13.2.8.1 **TGAB Pseudo-File Record Order.** Self-explanatory.

50.13.2.8.2 **TGAB Pseudo-File Field Structure.** Self-explanatory.

50.13.2.8.2.1 **TGAB Identifier Record.** Self-explanatory.

50.13.2.8.2.2 **File Name Record.** Self-explanatory.

50.13.2.8.2.3 **TGAB Header Record.** This mandatory record contains control information describing the TGAB. (Fields identified "Always Zero" are applicable to polygonized rather than gridded terrain. They have been included for consistency of format between the TPAB and TGAB area block headers.)

50.13.2.8.2.4 **Terrain Post Record.** Each of these records contains a single terrain elevation value from the grid of elevations making up the area block. The elevation posts are sequenced from the southwest corner of the area block to the northeast corner, with latitude intervals given from bottom to top along each longitude interval. The actual latitude and longitude intervals used are given in the parent TGAB Header record. Each elevation value will be represented as a coordinate in 3-D space. Coordinate triplets are given, rather than only elevation values, because elevation posts are located in the user-specified coordinate system, which may be different from geographic latitude/longitude.

50.13.2.8.2.5 **Pseudo-EOF Record.** Self-explanatory.

50.13.2.8.2.6 **Checksum Record.** This record contains the checksum value for the Terrain Grid Area Block Pseudo-File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the pseudo-file. The checksum does not include those characters contained in the Checksum record itself.

50.13.2.9 **Model Reference Area Block (MRAB) Pseudo-File.** The Model Reference Area Block Pseudo-File contains all model references included within an area block. The presence of the MRAB is optional depending on whether the user has requested models and whether available models are applicable to the area block. When present, the MRAB will follow the TGAB for the area block. Every model reference includes a model number which uniquely identifies a model within the model libraries. For audit trail purposes, each model reference also includes a feature number identifying the standard simulator data base feature which the model replaced. Each model is positioned within the area block by a coordinate defining the location of the model centroid. Each model reference also includes orientation angle, rotation, and scale factor descriptors. Project 2851 does not move culture to ensure that a model will not overlap a terrain polygon boundary. Such an overlap condition may cause difficulties for some image generators. Therefore, in GTDBs for which the user has requested culture fragmentation on terrain, a flag will be set in the model reference record to indicate the existence of an overlap, so that the user's Formatter Program can move the model if necessary.

50.13.2.9.1 **MRAB Pseudo-File Record Order.** Self-explanatory.

50.13.2.9.2 **MRAB Pseudo-File Field Structure.** Self-explanatory.

50.13.2.9.2.1 **MRAB Identifier Record.** Self-explanatory.

50.13.2.9.2.2 **File Name Record.** Self-explanatory.

50.13.2.9.2.3 **MRAB Header Record.** This mandatory record contains control information describing the file. It also contains a count of the individual model reference records which follow.

50.13.2.9.2.4 **Model Reference Record.** Self-explanatory.

50.13.2.9.2.5 **Pseudo-EOF Record.** Self-explanatory.

50.13.2.9.2.6 **Checksum Record.** This record contains the checksum value for the Model Reference Area Block Pseudo-File. This checksum is computed using a linear addition of the binary representations of all the ASCII characters contained in the pseudo-file. The checksum does not include those characters contained in the Checksum record itself.

50.13.2.10 **Area Block Pseudo-EOF Record.** It indicates that all pseudo-files describing a particular area block have been processed.

50.14 **Areal Texture (AT) Library.** There may be one Areal Texture Library (AT) following the SLAB. The AT is optional. When present, it contains all the areal textures referenced within the GTDB or explicitly requested by the user during GTDB generation. The AT begins with the NITF Header File. Following it are a set of files for each areal texture. If the texture is in Stage 1, then the files shall consist of the NITF Image Sub-Header (NITFISH) File and one or more files in the original format. These Original Format Image (OFI) Files may contain descriptive header-type data or the image itself, depending on the format and convention of the original source. The NITFISH File is used in this case to supplement this data with standard data that may be needed by the GTDB user (e.g., ground control points and tie points linking several images). For all other textures (those in a stage other than Stage 1), the files consist only of the NITFISH and the NITF Image Data (NITFID) File. The NITFISH is identical in format to the one used for Stage 1 textures; however, the fields that are required to hold valid data differs between the two. Naturally, the NITFID contains the binary image data itself. The AT contains all areal texture images included within the GTDB. The AT consists of files in the National Imagery Transmission Format (NITF) and files in their original native format. A description of NIFT can be found in the National Imagery Transmission Format, Version 1.1. (Application for copies may be addressed to Defense Intelligence Agency, DIA/DM - 1A, 3100 Clarendon Boulevard, Arelingon VA 22201-5317.) For ease of implementation, the AT format is identical to that of the Model Texture (MT) Library.

50.14.1 **AT File Structure.** Self-explanatory.

50.14.2 **AT Record Structure.** Self-explanatory.

50.14.2.1 **NITF Header File.** There shall be exactly one NITF Header (NITFH) File within the AT. While it is optional within a GTDB since the AT is optional, it is mandatory within the AT itself. It contains data used to identify the entire set of areal textures in the GTDB as well as a count of the number of areal textures and their individual sizes. Included in the standard NITFH data are fields for security. The file consists of both a standard NITF section as well as a GTDB-specific section. The NITF Header (NITFH) File is optional in a GTDB but mandatory in the AT Library. It exists in the AT if any areal texture is requested. The file contains information used to identify the entire set of areal images in the GTDB as well as a count of the number of areal images (textures) and their individual sizes. The format of this file is the same as that in the NITF standard. It is provided here for completeness and convenience. Each field has a label consisting of one to seven alphanumeric characters.

50.14.2.1.1 **Filename.** Self-explanatory.

50.14.2.1.2 **File Format.** Self-explanatory.

50.14.2.1.2.1 **GTDB User Defined Header Data.** The GTDB User Defined Header Data within the NITFH consists of GTDB- specific control data for texture. It follows the same conventions as the rest of the NITFH (i.e., field labels and end-of-line terminators).

50.14.2.1.2.1.1 **Image Tie Point Data.** Self-explanatory.

50.14.2.1.2.1.2 **Generic Texture Association Data.** Self-explanatory.

50.14.2.2 **NITF Image Sub-Header File.** There shall be exactly one NITF Image Sub-Header (NITFISH) File for each texture within the AT. It provides identification and descriptive information such as texture type, format, geographic location, environmental conditions, percentage of cloud cover, image quality ratings, resolution, bands, data sources, control points, tie points between images, footprints, neighbor texture identification, and sensor data. The file consists of both a standard NITF section as well as a GTDB-specific section. The NITF Image Sub-Header (NITFISH) File is optional in a GTDB but mandatory in the AT Library. It exists in the AT if any areal texture is requested. The file contains formatting control and other descriptive information for the texture following this file. The GTDB implementation of the NITF standard for the GTDB has some exceptions to the standard. These differences affect the Image Coordinate System Field, the Image Geographic Location Field, the use of Look Up Tables, and the image size. These modifications are explained here. While the NITF Image Coordinate System can be Universal Transverse Mercator (UTM), geodetic/geographic, geocentric, or none, the GTDB Image Coordinate System can be any of those as well as others. In the GTDB implementation, the coordinate systems can be any of the following: Geocentric, Mercator, Universal Transverse Mercator, Lambert, Polar, Local, Geodetic Float, Local Cartesian, or None. While NITF specifies a single character for this field, the GTDB implementation shall use the entire coordinate system name to specify it. The Image Geographic Location is limited in NITF to the nearest second in latitude and longitude. This may not be good enough for very high-resolution imagery, since one arc second spacing represents roughly 30 meters of ground resolution. Therefore, the format of the Image Geographic Location field has been changed, from NITF's four corner coordinates expressed to the nearest arc second, to four coordinates expressed in units of thousandths of arc seconds. It should be noted that the units may not be in geodeic coordinates, but rather metric units, if any projection other than geodetic is used. In this case, eleven digits are used (e.g. 1.234567890E+12). The exact outline is found in the Texture Footprint Data in the GTDB User Defined Image Data of the NITFISH. NITF supports the use of Look Up Tables (LUTs) with one byte of binary data per entry for image data. For visual texture, i.e., color or intensity data, the GTDB NITFISH shall not use LUTs. All such data shall be directly stored in the NITF Image Data File. This is fully compliant with NITF since the number of LUTs can be zero. For SMC/FDC data, LUTs may or may not be used, depending on the user's preference; however, if LUTs are used, the LUT entry shall be entirely in ASCII with a length of seven bytes. The first two bytes represent the SMC (0 - 15), while the following five bytes represent the ASCII FDC value. NITF limits the number of pixels per image to 4096 in the horizontal direction and 7700 in the vertical, with a maximum of 16 bits per pixel per band. In order to more fully support GTDB users, the GTDB will not observe this limitation. The image size will have no logical limitation. The format of this file is the same as that in the NITF standard. It is provided here for completeness and convenience. Each field has a label consisting of one to seven alphanumeric characters.

50.14.2.2.1 **Filename.** For example, the first areal NITF Image Sub-Header File would be named 'TEXA00001.HDR'.

50.14.2.2.2 **File Format.** Self-explanatory.

50.14.2.2.2.1 **GTDB User Defined Image Data.** The GTDB User Defined Image Data within the NITFISH contains formatting control and other descriptive information for the image following the GTDB User Defined Image Data Record. This set of fields is defined specifically for the GTDB and supplements the standard fields provided in the NITF Image Sub-Header Record. The GTDB User Defined Image Data follows the same conventions as the rest of the NITFISH (i.e., field labels and end-of-line terminators).

50.14.2.2.2.1.1 **General Processing Data.** Self-explanatory.

50.14.2.2.2.1.2 **Source Data.** Self-explanatory.

50.14.2.2.2.1.3 **Environmental Conditions Data.** Self-explanatory.

50.14.2.2.2.1.4 **Texture Footprint Data.** Self-explanatory.

50.14.2.2.2.1.5 **Neighbor Texture Association Data.** Self-explanatory.

50.14.2.2.2.1.6 **Model Association Data.** Self-explanatory.

50.14.2.2.2.1.7 **Image Control Data.** Self-explanatory.

50.14.2.2.2.1.8 **Sensor Image Data Descriptor Data.** The usefulness of this data is dubious. Until experience is gained in its use, the value will remain questionable. It may be found that this data should be removed because it is too difficult to acquire, or because sensors are too varied to standardize on a set of parameters to describe them.

50.14.2.3 **Original Format Image File(s).** For each Stage 1 areal texture, there will be one or more Original Format Image (OFI) Files following the NITFISH. The OFI is mandatory for a Stage 1 texture image. When present, it contains the original identification, descriptive information, and image itself in its original native format for a specific Stage 1 texture. The OFI files are the original source files provided. Their content and format are source dependent. Their filenames are modified, while their original source filenames are provided in the TLH. The Original Format Image (OFI) File is optional in a GTDB. It exists if the texture requested is Stage 1 areal texture. The OFI consists of the original data provided by an image source such as EOSAT. The OFI includes all descriptive header information as well as the imagery provided by the source in its original format. The header information and the image itself are not altered in the GTDB. Since the structure of the file(s) is source-dependent, there is no description of such files in this document. The original filename has been replaced by a filename following GTDB conventions in order to ensure uniqueness of filenames within a GTDB. The original filename is associated with GTDB filename within the TLH.

50.14.2.3.1 **Filename.** For example, assume three files existed for a single Stage 1 texture (image) which was the first areal texture within the GTDB. The three OFI Files would be named, in order of their appearance within the GTDB, 'OFIA00001.A', 'OFIA00001.B', and 'OFIA00001.C'.

50.14.2.4 **NITF Image Data File.** There shall be exactly one NITF Image Data (NITFID) File for each texture not in Stage 1 following the NITFISH within the AT. It provides the actual image data itself. While NITF supports the use of Look-Up Tables (LUTs), the GTDB implementation shall store the texel values directly in the image data without the use of LUTs for all visual color and intensity texture; however, for SMC/FDC data, LUTs may be used depending on the GTDB parameters. The GTDB will use the future multi-block image format outlined in the NITF standard; while it is not currently supported by NITF, it should be adopted in the future. Its current adoption will enable the GTDB to offer great formatting flexibility. The NITF file supports storage of texture images at any resolution and at any rectangular size. These shall be determined by GTDB parameters. Image compression is supported by NITF, but will not be initially implemented in the GTDB. With the user defined fields, the NITF will support all coordinate systems currently supported in the GTDB. The NITF Image Data File is optional within a GTDB. It exists if the texture requested is non-Stage 1 areal texture. Images may contain RGB, grayscale, or SMC/FDC data. The future multiblock image format presented in that document has been implemented in this record. Under this format, image data may be formatted on a pixel-by-pixel, line-by-line, or block-by-block basis with bands either sequential or interleaved between pixels, lines, or blocks. "Image data within a block shall be formatted on a row by row basis, from left to right along each row or line, and from the top of the block to the bottom, down the rows. Data shall begin with the N bits of pixel (0,0) (the first row, first column) of the first block." The NITF image coordinate system starts with (0,0) at the upper left corner of the image, with the first coordinate increasing from top to bottom, and the second coordinate increasing from left to right. "The N bits of each pixel shall be in order beginning with the most significant bit (MSB) and ending with the least significant bit (LSB). This is followed by the N bits of data for pixel (0,1), which is the first row, second column of the first block. The N bits of data for pixel (1,0) (the first column of the second row) of the first block shall follow the last pixel of the first row of the first block. The MSB of data for the first pixel of the first line of the second block shall follow the LSB of data for the last pixel of the last line of the preceding block. The end of the image data shall be LSB of the N bits of the last pixel, last row, last block of the last band." "In Sequential Image Mode (i.e., Band Sequential), all of the blocks of the first band are followed by all of the blocks by the second band, and so on. Thus, the first block of the first band is followed by the data for the second block of the first band. The last block of the first band is then followed by the first block of the second band. In Interleaved Image Mode (i.e., Band Interleaved or Pixel Interleaved), the first block of the first band is followed by the first block of the second band which is then followed by the first block of each subsequent band. The first block of the last band is followed by the second block of the first band, and so on."

50.14.2.4.1 **Filename.** For example, the NITF Image Data File corresponding to the first areal texture (and thus, to the first areal NITF Image Sub-Header File) would be named 'TEXA00001.DAT'.

50.14.2.4.2 **File Format.** Self-explanatory.

50.15 **Model Texture (MT) Library.** There may be one Model Texture Library (MT) following the AT. The MT is optional. When present, it contains all the model textures referenced within the GTDB or explicitly requested by the user during GTDB generation. The MT begins with the NITF Header File. Following it are a set of files for each model texture. The files and their usage are identical to that of the AT. The MT contains all model texture images included within the GTDB. A description of each of the NITF files follows; for more details, one should consult the NITF document(s). A description of files in their original native format can be found in the appropriate documentation. For ease of implementation, the MT format is identical to that of the Areal Texture (AT) Library.

50.15.1 **MT File Structure.** Self-explanatory.

50.15.2 **MT Record Structure.** Self-explanatory.

50.15.2.1 **NITF Header File.** There shall be exactly one NITF Header (NITFH) File within the MT. While it is optional within a GTDB since the MT is optional, it is mandatory within the MT itself. The file format is identical to that in the AT. The differences between the AT NITFH and the MT NITFH are in which fields are required to contain valid real data and which fields contain null data. The NITF Header (NITFH) File is optional in a GTDB but mandatory in the MT Library. It exists in the MT if any model texture is requested. The file contains information used to identify the entire set of model images in the GTDB as well as a count of the number of model images (textures) and their individual sizes.

50.15.2.2 **NITF Image Sub-Header File.** There shall be exactly one NITF Image Sub-Header (NITFISH) File for each texture within the MT. The file format is identical to that in the AT. The differences between the AT NITFH and the MT NITFH are in which fields are required to contain valid real data and which fields contain null data. The NITF Image Sub-Header (NITFISH) File is optional in a GTDB but mandatory in the MT Library. It exists in the MT if any model texture is requested. The file contains formatting control and other descriptive information for the texture following this file.

50.15.2.3 **Original Format Image File(s).** For each Stage 1 model texture, there will be one or more Original Format Image (OFI) Files following the NITFISH. The OFI is mandatory for a Stage 1 texture image. When present, it contains the original identification, descriptive information, and image itself in its original native format for a specific Stage 1 texture. The OFI files are the original source files provided. Their content and format are source dependent. Their filenames are modified, while their original source filenames are provided in the TLH. The Original Format Image (OFI) File is optional in a GTDB. It exists if the texture requested is Stage 1 model texture. The OFI consists of the original data provided by an image source such as EOSAT. The OFI includes all descriptive header information as well as the imagery provided by the source in its original format. The header information and the image itself are not altered. Since the structure of the file(s) is source-dependent, there is no description of such files in this document.

50.15.2.4 **NITF Image Data File.** There shall be exactly one NITF Image Data (NITFID) File for each texture not in Stage 1 following the NITFISH within the MT. It provides the actual image data itself. The file format is identical to that in the AT. It supports all of the general capabilities supported by the NITFID in the AT. The NITF Image Data File is optional within a GTDB. It exists if the texture requested is non-Stage 1 areal texture. Images may contain RGB, grayscale, or SMC/FDC data. The future multiblock image format presented in that document has been implemented in this record. Under this format, image data may be formatted on a pixel-by-pixel, line-by-line, or block-by-block basis with bands either sequential or interleaved between pixels, lines, or blocks.

50.16 **Microdescriptors.** Microdescriptors are a special class of data structures developed by the Defense Mapping Agency (DMA) to encode complex attributes in its digital cartographic data bases. Their use has so far been limited to prototype data bases (such as Level V Digital Feature Analysis Data (DFAD)). However, they offer a potentially useful way to capture complex information which would be very useful for high-resolution simulation. Microdescriptors have been specified for three general classes of data: (a) They can be used to provide additional details on the subcomposition of large or composite features without going through the effort of digitizing and individually attributing the subdetail. Microdescriptors of this type which will be supported by Project 2851 include the Homogeneous Area Microdescriptor, the Pattern Distribution Microdescriptor, and the Vertically Composite Microdescriptor. These microdescriptors support simulation of low altitude radar returns by providing a pattern of structures for modeling of vertical reflecting surfaces and culture masking effects. This type of microdescriptor could be useful in multi-sensor simulator applications by supporting more realistic and better correlated synthetic breakup, model design, and selection of areal photo texture. (b) Microdescriptors can also be used to associate data of a functional or operational nature with culture features. Microdescriptors of this type to be supported by the GTDB include the Drainage Microdescriptor, the Transportation Microdescriptor, and the Vegetation Microdescriptor. This type of microdescriptor could be useful in simulator applications to support more realistic models and scenario effects. Some of the attributes (e.g., SMC of river banks) are important for low altitude radar simulation. (c) For the GTDB, a new class of microdescriptors for the purpose of encoding temporal effects. This Temporal Effects Microdescriptor will support descriptions of how culture characteristics will change based on time of day, season, weather, and other conditional situations. This type of microdescriptor could support more realistic and better correlated scenario effects. One possible application in radar simulation would be establishing local seasonal and climate thresholds for changes in water body states. In the following subparagraphs, each type of microdescriptor supported by the GTDB is described using the classic record and field structure. This convention is used for clarity in describing the microdescriptors and for consistency with normal applications of microdescriptors. However, the GTDB will actually store each microdescriptor attribute as an individual record within a microdescriptor record-type. This means that a Homogeneous Area (HA) microdescriptor, for example, would be stored as three records, one for each attribute field associated with the HA microdescriptor. The general structure of the microdescriptor records stored within a GTDB is to have a Microdescriptor Type Field, which identifies the microdescriptor and the particular attribute being

described, followed by a data field which contains the data value for that particular microdescriptor attribute. This approach was taken to give the GTDB the flexibility to accommodate new or modified microdescriptors in the future without changing the data base design. An additional benefit is that storage space is used only for attributes for which values have been captured.

50.16.1 **GTDB Microdescriptor Records.** Self-explanatory.

50.16.1.1 **Homogeneous Area Microdescriptor Record (HA).** The Homogeneous Area Microdescriptor describes the basic size and material of a standard subunit of a culture feature captured as one homogeneous area.

50.16.1.2 **Pattern Distribution Microdescriptor Record (PM).** The Pattern Distribution Microdescriptor provides parameters describing the distribution of subfeatures within a larger areal feature. This microdescriptor may be used to support automated algorithms for synthetic breakup.

50.16.1.3 **Drainage Microdescriptor Record (TTAD).** The Drainage Microdescriptor provides operationally useful information about rivers.

50.16.1.4 **Transportation Microdescriptor Record (TTAT).** The Transportation Microdescriptor provides operationally useful information about bridges, highways, and railroads.

50.16.1.5 **Vegetation Microdescriptor Record (TTAV).** The Vegetation Microdescriptor provides operationally useful information about soil and undergrowth conditions.

50.16.1.6 **Vertically Composite Microdescriptor Record (VC).** The Vertically Composite Microdescriptor describes the vertical components of a structure which has been captured as a single culture feature but which in fact consists of objects of interest stacked on top of other objects of interest.

50.16.2 **Temporal Effects Microdescriptor Records.** The Temporal Effects Microdescriptors provide situation-dependent alternate attributes for a feature which has been described in the data base as it would appear under nominal conditions. The GTDB storage structure for these microdescriptors is the same as for the other microdescriptors, except that one additional field has been added to count the number of attributes which are affected by the temporal condition. Several specific microdescriptor types have been defined to address various temporal conditions affecting simulation. These formats are described below.

50.16.2.1 **Weather Effects Microdescriptor Record (TEW).** The TEW record is used to describe weather conditions which trigger changes to one or more attributes describing a feature or model.

50.16.2.2 **Seasonal Effects Microdescriptor Record (TES).** The TES record is used to describe a season which triggers changes to one or more attributes describing a feature or model.

50.16.2.3 **Time of Day Microdescriptor Record (TET).** The TET record is used to describe a time-of-day threshold which triggers changes to one or more attributes describing a feature or model.

50.16.2.4 **Ground Conditions Microdescriptor Record (TEG).** The TEG record is used to describe ground conditions which trigger changes to one or more attributes describing a feature or model.

50.16.2.5 **Alternate Attributes Record (TEAA).** Each TEAA record is used to describe the alternate value of an attribute describing a feature or model.

50.17 **Feature Codes and Attributes.** The Feature Attribute Coding Standard (FACS) is being developed by the Defense Mapping Agency (DMA) to standardize data collection and attribution guidelines in its digital cartographic data bases. The use of FACS in current DMA products has so far been limited, but DMA has made a commitment to the adoption of FACS in future products currently under development. A standard such as FACS is clearly necessary if various data bases dealing with geographic/cartographic data are to be successfully integrated for advanced applications. The GTDB will use FACS standards in two general ways: (a) First, it will use the hierarchical feature categories defined in FACS to identify what a feature is. These identifiers will be stored in a field called the Feature Descriptor Code (FDC) present in every feature record. The FACS feature categories are generally a superset of categories used by older DMA standards such as Digital Feature Analysis Data (DFAD). However, there are some feature types used in the training simulation community which are not presently included in FACS. The GTDB has defined additional FACS-like FDCs to represent these features. A complete list of FDCs supported by the GTDB is given in the separately-bound Appendix B to this standard. (b) Second, the GTDB has adopted a FACS-like approach to feature attribution as a technique which will allow the future addition of an indefinite number of new attributes without having to modify the data base structure. Every feature file in the data base contains an optional record type called a FACS Record. Each occurrence of a FACS Record will be a feature attribute not contained among the "core" attributes already defined in the feature records. The internal field structure of the FACS Record has been generalized so that it contains an attribute identifier, as well as an attribute value. Thus, as new attribution requirements are identified, the additional attributes can be given unique attribute identifiers and stored sequentially among the FACS records.

60 NOTES

60.1 **Intended use.** This appendix is intended to be used as a guide for the interpretation of the content of this Generic Transformed Data Base standard.

60.3 **Subject term (key word) listing.**

Data Base

60.4 **Referenced documents.** The following documents were used as references, in preparation of this standard.

ANSI/MIL-STD-1815A, Ada Programming Language.

Defense Mapping Agency Digital Landmass System Product Specification, First Edition, July 1977.

Defense Mapping Agency Digital Landmass System Product Specification, Second Edition, April 1983.

Defense Mapping Agency Product Specifications for a Prototype Data Base to Support High Resolution Sensor Simulation, First Edition, December 1979.

Defense Mapping Agency Prototype High Resolution Data Base Product Specification, First Edition, August 1980.

Defense Mapping Agency Level X Product Specification, First Edition, June 1983.

Defense Mapping Agency Feature File Product Specification, First Edition, August 1984.

Defense Mapping Agency Standard Linear Format Product Specification, First Edition.

Defense Mapping Agency (DMA) High Resolution Data (Level X) Specification for B-1B Simulator.

DMA Standard Supporting Mark 90, Section 100, Glossary of Feature/Attribute Definitions, Second Edition, June 1988, revised December 1988.

60.5 **Design Notes.** The following is an overview of design features of the GTDB.

60.5.1 For configuration management and data base verification purposes, the design includes storage of all transformation parameters used to generate the GTDB in the Gaming Area Header File.

60.5.2 The GTDB includes a class of header data records to provide data base statistics for the user. These statistics include measures of data density and distribution for different area blocks and simulator levels of detail (SLODs) within a GTDB. These statistics are intended to be helpful to users in planning and optimizing the exploitation of GTDBs.

60.5.3 When polygonized (rather than gridded) terrain is requested, there are optional fields which will associate each terrain polygon with all culture features lying upon it. If culture fragmentation along terrain polygon boundaries is requested during transformation, then the individual fragments will be tied to the terrain polygons.

60.5.4 As originally designed, the GTDB carried a security level attribute all the way down to the individual feature and model level. Although the Project 2851 data base concept continues to anticipate that a GTDB may be created from data of varying security levels, it is impractical to expect GTDB recipients to control secure data within an integrated data base at extreme levels of granularity. Therefore, the effective security level was changed to the file level within the GTDB. If any data within a GTDB file is classified, then the entire file is classified to the highest level of any data within it. Technically, a GTDB may consist of some files which are classified and some which are not, but the actual classification will be dependent upon operational implementation.

60.5.5 There will be users who wish to maintain a high level of correlation between two separate GTDBs built to support two different simulators; e.g., between radar and visual displays, or among networked simulators. Perfect correlation between dissimilar simulators systems is not always possible. However, Project 2851 has included design features which will help maximize correlation. First, fields have been added which may be used to flag and prioritize data base features which are particularly significant for correlation. Second, for users who require gridded terrain, there is an option for deriving the grid values from the polygonized terrain of related (or the same) GTDB.

60.5.6 A class of culture microdescriptors has been designed specifically to encode temporal characteristics. These new microdescriptors can be used to help standardize simulator rendering of seasonal, weather, time-of-day, and conditional effects.

60.5.7 Attributes which may be sensor-specific (e.g., directivity) are carried as dual attributes (e.g., radar directivity and infrared directivity). Attributes which do not vary with the sensor (e.g., height or surface material category) are included only once.

60.5.8 The sequence of files within the GTDB has been designed to minimize the probable volume of processing by a user's Formatter Program (FP). As a general rule, referenced data items are presented prior to their being referenced.

60.5.9 The GTDB currently has all data represented in the ASCII character set rather than in VAX numeric formats or Ada data types. This makes the tapes more universally readable without special programming for system-specific data conversions. Due to the fact that ASCII data sets tend to become quite large, however, this may be changed at some future date.

60.5.10 Each file within the GTDB begins with a file identifier record and a filename record.

60.5.11 Consolidated vertex files have been defined to reduce redundant specification of coordinates. There are consolidated terrain/culture vertex file on a per area block basis.

60.5.12 Vertex normals can be included as a data base option. These normals apply to polygonized terrain and models.

60.5.13 Data fields are separated by the ASCII null character.

60.5.14 An effort has been made to reduce the number of separate physical files used to store the GTDB data. Data units called 'pseudo-files' have been defined for collections of records which can logically be treated as independent files but which, for performance reasons, have been physically grouped with other pseudo-files within a larger physical file. The grouping of psuedo-files into fewer physical files has been found to improve I/O performance dramatically.

60.5.15 The Culture Reference Record within the Terrain Polygon Area Block Pseudo-File has been designed to make it possible to associate model references, as well as culture feature references, with a given terrain polygon.

60.5.16 Within the Terrain Grid Area Block Pseudo-File, terrain elevation post values are stored one post per record. This approach reflects the wide variation possible in area block sizes.

60.5.17 Restrictions on area block size are that no side may exceed 15 minutes of arc, and that neighboring area blocks must share mutually identical boundaries.

STANDARDIZATION DOCUMENT IMPROVEMENT PROPOSAL

INSTRUCTIONS

1. The preparing activity must complete blocks 1, 2, 3, and 8. In block 1, both the document number and revision letter should be given.
2. The submitter of this form must complete blocks 4, 5, 6, and 7
3. The preparing activity must provide a reply within 30 days from receipt of the form

NOTE: This form may not be used to request copies of documents, not to request waivers, or clarification of requirements on current contracts. Comments submitted on this form do not constitute or imply authorization to waive any portion of the referenced document(s) or to amend contractual requirements.

I RECOMMEND A CHANGE:	1. DOCUMENT NUMBER MIL-STD-1820	2. DOCUMENT DATE *(YYMMDD)* 17 December 1992

3. DOCUMENT TITLE
Generic Transformed Data Base Design Standard

4. NATURE OF CHANGE *(Identify paragraph number and include proposed rewrite, if possible. Attach extra sheets as needed.)*

5. REASON FOR RECOMMENDATION

6. SUBMITTER

a. NAME *(Last, First, Middle Initial)*	b. ORGANIZATION	
c. ADDRESS *(Include Zip Code)*	d. TELEPHONE *(Include Area Code)* (1) Commercial (2) AUTOVON *(If applicable)*	7. DATE SUBMITTED *(YYMMDD)*

8. PREPARING ACTIVITY

A. NAME ASC/ENES	B. TELEPHONE *(Include Area Code)* (1) Commercial 513-255-6281 (2) AUTOVON *(If applicable)* 785-6281
C. ADDRESS *(Include Zip Code)* Bldg 126, 2664 Skyline Drive Wright-Patterson AFB OH 45433-7800	IF YOU DO NOT RECEIVE A REPLY WITHIN 45 DAYS, CONTACT Defense Quality and Standardization Office 5203 Leesburg Pike, Suite 1403, Falls Church VA 22041-3466 Telephone (703) 756-2340 AUTOVON 289-2340

DD Form 1426, OCT 89 Previous editions are obsolete 198/290

NOT MEASUREMENT SENSITIVE

MIL-STD-1821
17 June 1993

MILITARY STANDARD

STANDARD SIMULATOR DATA BASE (SSDB) INTERCHANGE FORMAT (SIF) DESIGN STANDARD

AMSC No. N/A

FSC: 69GP

Distribution Statement A: Approved for public release; distribution is unlimited.

DEPARTMENT OF DEFENSE
WASHINGTON DC 20301-8000

STANDARD SIMULATOR DATA BASE (SSDB) INTERCHANGE FORMAT (SIF)

1. This military standard is approved for use by all Departments and Agencies of the Department of Defense.

2. Beneficial comments (recommendations, additions, deletions) and any pertinent data which may be of use in improving this document should be addressed to: ASC/ENOS, Wright-Patterson AFB OH 45433-7800, by using the self-addressed Standardization Document Improvement Proposal (DD Form 1426) appearing at the end of this document, or by letter.

FOREWORD

1. The purpose of this standard is to define the Standard Simulator Data Base (SSDB) Interchange Format (SIF).

2. The Standard Simulator Data Base (SSDB) is the central repository of validated simulator databases for the DoD training simulation community. The SSDB was developed under an Air Force program called Project 2851 (contract F33657-86-C-0182). Tri-Service coordination has been maintained via the Joint Technical Coordinating Group for Training Systems and Devices (JTCG-TSD). The SSDB will be maintained at the DoD Simulator Data Base Facility (SDBF).

3. The SIF will serve as an input/output vehicle for sharing digital simulator databases via the SSDB. Database builders may be tasked to supply their databases to the SDBF in this format. The SDBF would then be responsible for integrating the databases into the SSDB, from which data may later be extracted for use by other simulator systems, in either SIF or Generic Transformed Data Base (GTDB) format. This will allow the Government to re-use databases created for specific simulation programs.

4. This standard defines two different versions of SIF, in order to support two different scenarios for sharing of SSDB data. In one form, the SSDB can be made available in its complex internal system-specific format to support distributed maintenance by SDBF-compatible data base systems, or to support autonomous but SSDB-compatible data base production by training system programs. In SIF's other form, there will be a mechanism for passing the essential contents of simulator data bases, including the SSDB, for use or maintenance on systems with significantly different software than the SDBF. The "SSDB Interchange Format for Distributed Processing (SIF/DP)" has been defined to handle the first situation, while the "SSDB Interchange Format for High Detail Input/Output (SIF/HDI)" has been defined for the second.

5. The two SIF formats, SIF/DP and SIF/HDI, are logically identical but differ in physical format. Use of one format where the other would be more appropriate is likely to cause the implementing program to incur unnecessary costs. It is therefore important that the SIF user possess an understanding of the distinction between the two alternatives, before specifying either for use.

6. When it is required that a simulator program receive database inputs from the SDBF, a third format may be used. This third format designated the Generic Transformed Data Base (GTDB), is documented in MIL-STD-1820. The GTDB is strickly an output product format of the SDBF and is a more thoroughly processed database than SIF. It is capable of supporting a much greater range of data selection and formatting options than either SIF/HDI or SIF/DP. GTDB should be considered for use instead of SIF when the receiving system wishes to minimize post-processing of the data.

CONTENTS

CONTENTS

CONTENTS

CONTENTS

CONTENTS

CONTENTS

CONTENTS

CONTENTS

CONTENTS

FIGURES

APPENDICES

1. SCOPE

1.1 Scope. This standard defines the Standard Simulator Data Base Interchange Format (SIF).

1.2 Applicability. This standard should be invoked when the acquisition agency establishes a requirement that a digital database created to support a specific training simulator program be shared with other programs via the Simulator Data Base Facility (SDBF) Standard Simulator Data Base (SSDB), and/or that a digital database available from the SSDB be used as an input source by the invoking program.

1.3 Application guidance. The SIF standard encompasses two independent data formats, SIF for High Detail Input/Output (SIF/HDI) and SIF for Distributed Processing (SIF/DP). The acquisition agency should select the most appropriate alternative based on the requirements and constraints of a particular program. In selecting the format most applicable to a particular program, a third format, GTDB, should be considered as well. The selection of a standard format should be based upon the following general criteria.

1.3.1 SIF/HDI. SIF/HDI is designed to serve as a comprehensive set of formats for exchange of simulator databases between the SDBF and external database generation/transformation systems. It may be used to transmit a validated database to the SDBF for storage in the SSDB central repository and subsequent dissemination to other programs. It may also be used to receive and input a database from the SDBF SSDB for further processing on a simulator database generation system. SIF/HDI should be specified when the invoking program wishes to export a database to the SDBF and/or to import a database contained within the SSDB.

1.3.2 SIF/DP. SIF/DP is designed for exchange of databases using formats essentially identical to internal binary formats maintained on the SDBF system. It is the preferred format for distributed or supplemental maintenance and enhancement of internal SSDB files. Systems processing SIF/DP data would require SDBF-compatible hardware and software.

1.3.3 GTDB. The Generic Transformed Data Base (GTDB) format, MIL-STD-1820, is strickly an output product format from the SDBF. It is capable of a much greater degree of tailoring than SIF, in terms of data content, encoding formats, and degree of transformation processing. It is the preferred SDBF product format when the recipient wishes to minimize additional filtering and/or transformation of the data. The GTDB format is not supported as a SDBF data source; therefore, it should not be specified when the invoking program wishes to export a database to the SDBF.

1.4 Tailoring of requirement descriptions. The detailed technical requirements of this format have been structured to permit tailoring to suit the particular database requirements of an individual program. Under normal circumstances, it should be sufficient for an acquisition agency to specify compliance with the SIF standard as a whole, with specific exceptions granted on a case-by-case basis with the concurrence of the SDBF. First-time users of the SIF standard should read Appendix C for general guidance on applying the standard to particular applications.

1.5 Method of reference. This standard should be invoked by requiring that a program utilize and/or deliver databases in accordance with MIL-STD-1821. Interface with the SDBF is implicit in any invocation of this standard.

2. APPLICABLE DOCUMENTS

2.1 Government documents

2.1.1 Specifications, standards, and handbooks. The following specifications, standards, and handbooks form a part of this document to the extent specified herein. Unless otherwise specified, the issues of these documents shall be those listed in the issue of the Department of Defense Index of Specifications and Standards (DoDISS) and supplement hereto, cited in the solicition (see 6.2).

Military Standard

MIL-STD-1820 Generic Transformed Data Base Design Standard

(Unless otherwise indicated, copies of federal and military specifications, standards, and handbooks are available from the Naval Publications and Forms Center, ATTN: NPODS, 5801 Tabor Avenue, Philadelphia PA 19120-5099.)

2.1.2 Other Government documents, drawings, and publications. The following other Government documents, drawings, and publications form a part of this document to the extent specified herein. Unless otherwise specified, the issues shall be those cited in the solicitation (see 6.2)

DEFENSE INTELLIGENCE AGENCY

DDM-2600-63220-90	National Imagery Transmission Format (NITF), Version 1.1, 1 March 1989, sections 1 through 4.5

(Application for copies should be addressed to Defense Intelligence Agency, DIA/DM-1A, 3100 Clarendon Boulevard, Arlington, VA 22201-5317.)

TASC

U-321/CIO-2 MIL-STD-XXX-BWC3	"Joint Photographic Experts Group (JPEG) Image Compression For The National Imagery Transmission Format Standard", Director of Central Intelligence

(Application for copies should be addressed to TASC, ATTN: NTB Secretary, 55 Walkers Brook Drive, Reading MA 01867-3297.)

U.S. DEPARTMENT OF COMMERCE

	Initial Graphics Exchange Specification (IGES), Version 4.0, June 1988, sections applicable to Constructive Solid Geometry (CSG)

(Application for copies should be adressed to U.S. Department of Commerce, National Bureau of Standards.)

2.2 Non-Government publications. The following documents form a part of this document to the extent specified herein. Unless otherwise specified, the issues of the documents which are DoD adopted shall be those listed in the issue of the DoDISS cited in the solicitation (see 6.2). Unless otherwise specified, the issues of documents not listed in the DoDISS shall be the issues of the documents cited in the solicitation (see 6.2).

AMERICAN NATIONAL STANDARDS INSTITUTE (ANSI)

ANSI X3.4	American Standard Code for Information Interchange (ASCII)
ANSI X3.27	Information Systems - File Structure and Labeling of Magnetic Tapes for Information Interchange
ANSI/IEEE STD 754	Binary Floating Point Arithmetic

(Application for copies should be addressed to American National Standards Institute, 11 West 42nd Street, New York NY 10036.)

(Non-Government standards and other publications are normally available from the organizations that prepare or distribute the documents. These documents also may be available in or through libraries or other informational services.)

2.3 Order of precedence. In the event of a conflict between the text of this document and the references cited herein, the text of this document shall take precedence. Nothing in this document, however, supersedes applicable laws and regulations unless a specific exemption has been obtained.

3. DEFINITIONS AND ACRONYMS

3.1 Definitions. For the purpose of this standard, the following definitions shall apply.

3.1.1 Data Fields. The definitions of all data fields used in SIF are provided in Appendix A, SIF Data Dictionary.

3.1.2 Files and Records. A description of the application of each file and record type may be found in Appendix C, Rationale and Guidance, Section 50.

3.1.3 Terms. As used in this document, the following terms are defined as shown.

Areal Feature. The representation of an object in a culture data base as a closed polygon, with associated attributes.

Constructive Solid Geometry (CSG). A method of representing three-dimensional objects in which complex shapes are created through the additive and subtractive combination of volumetric primitives, such as cylinders, spheres, and prisms.

CSG Model. A model created using CSG techniques.

Culture Data. A two-dimensional digital data set containing information representing both natural and manmade features on the Earth's surface.

Data Base Generation System (DBGS). A hardware/software system used to transform raw hardcopy and softcopy sources (such as maps and imagery) into composite digital data bases, which are subsequently used in real-time simulator image generation equipment anf supports constructive (2D) modelling and simulation.

Digital Feature Analysis Data (DFAD). A standard DMA data base consisting of selected natural and manmade planimetric features, classified as point, line, or areal features as a function of their size and composition. DFAD is stored in a spaghetti vector format.

Digital Terrain Elevation Data (DTED). A standard DMA data base consisting of a uniform matrix of terrain elevation values.

Distributed Processing (DP). One concept of SIF production, wherein alternate production centers are established at remote facilities, are equipped with SDBF-compatible DBGSs, and exchange SSDB files directly with the SDBF.

Edge. The line formed by the connection of two vertices.

Elevation Data. Digital information representing the variation in elevation of the Earth's surface, relative to mean sea level.

Face. A planar two- or three-dimensional structure formed by the closed connection of a series of segments.

Face-Based Texture. A technique for applying a texture map to a single polygon, wherein the texture pattern is of a fixed size, and replicated as often as is necessary to cover the entire polygon.

Feature Attribute Coding Standard (FACS). A Defense Mapping Agency developed system of alphanumeric codes, which are used to represent various properties of cultural features stored in a data base.

Feature Data. Same as Culture Data.

Feature Descriptor Code (FDC). An alphanumeric code used to identify the type of a cultural feature stored in a data base.

Generic Texture. A file containing a non-geospecific pattern, eligible for mapping repeatedly onto any polygon in the data base.

Generic Transformed Data Base (GTDB). A product of the SDBF, consisting of data which has been extracted from the SSDB and tailored to meet the specific characteristics of a particular training simulator image generator and/or constructive 2D M&S.

Global-Based Texture. A technique for applying texture to terrain, wherein an orthorectified image is mapped onto the terrain polygons at the corresponding geographic location.

Gridded Data. Digital information which is uniformly distributed in the form of a two-dimensional matrix, where a data value is provided for each (x,y) coordinate. Both terrain elevation and rasterized texture are considered types of gridded data files.

High Detail Input/Output (HDI). A concept of SIF production, wherein external producers use non-SDBF-compatible DBGSs to create SIF data sets for SDBF consumption; and conversely, the SDBF provides SIF data sets to external consumers for application on training simulators. The HDI term derives from the fact that the primary purpose of this interface is to facilitate the reuse of densely-populated data bases, which can be quite expensive to create.

Initial Graphics Exchange Standard (IGES). A format for the distribution of computer graphics files developed by the National Institute of Science and Technology. IGES is used as the basis of the SIF Constructive Solid Geometry (CSG) model format.

Island. A section of high-resolution data embedded in a data base LOD of lower overall resolution. An island is also known as hi-res patch.

Level of Detail (LOD). One of a set of representations of a given unit of terrain, culture, texture, or model data, which contains only those components of the unit which can be independently defined at a specified resolution.

Lineal Feature. The representation of an object in a culture data base as a vector or connected series of vectors, with associated attributes.

Linear Feature. Same as Lineal Feature.

Manuscript. A geographic subset of the total data base, which has been physically segregated for purposes of manageability.

Model. A two-dimensional or three-dimensional geometric representation of a physical entity, which includes sufficient attribution to present a recognizable portrayal of that object when rendered on a real-time image generator.

Model-Based Texture. A technique for applying a texture map to a model, wherein the texture pattern is applied to multiple polygons simultaneously.

National Imagery Transmission Format (NITF). A digital file format designed by the Defense Intelligence Agency for the distribution of raster imagery data.

Node. The coordinates specifying a point location in a two-dimensional plane, or three-dimensional space.

Non-Mapped Texture. The transmittal of texture patterns in a SIF data set without associating them with any models or features.

Planimetric Data. Same as Culture Data.

Point Feature. The representation of an object in a culture data base as a single point in space, with associated attributes.

Point Light Feature. The representation of a light-emitting point feature in a culture data base.

Point Light String Feature. The representation of a series of logically related point light features in a culture data base.

Polygon. Same as Face.

Polygonal Model. A model created through the definition of boundary polygons, which implicitly define solid objects.

Raster Data. A matrix of evenly spaced rows and columns of texture or picture elements (texels or pixels). Examples and SPOT and LANDSAT imagery.

Rendering Priority. The relationships established among objects such as to define which occult the others from different perspectives.

Segment. A series of connected edges.

Separating Plane. A non-displayed plane, which may be defined by a polygon, inserted into a three-dimensional model for the purpose of establishing priority cells or clusters, which speeds the computation of hidden surfaces on the model on certain image generators.

Separation Plane. Same as Separating Plane.

SIF Consumer. A training simulator contractor with the requirement to utilize data bases in SIF format.

SIF Producer. A training simulator contractor with the requirement to deliver digital data bases to the Government in SIF format.

SIF/DP. The version of SIF designed to support the Distributed Processing scenario.

SIF/HDI. The version of SIF designed to support the High Detail Input/Output scenario.

Simulator Data Base Facility (SDBF). The production and maintenance center for DoD training simulator data bases, to be managed by the Defense Mapping Agency, and located in St. Louis, MO.

Surface Material Category (SMC)/FDC Texture. A Stage 3 Specific Areal Texture file for which surface material and feature descriptor codes have been substituted for the raw intensity value of each pixel.

"Spaghetti" Vector. A vector data format wherein features are independently defined, nodes and edges are associated with individual features, and no topological relationships are maintained among features.

Stage 1 Specific Areal Texture. A texture file which contains raw digital imagery (i.e., that which has not been modified through any image processing technique), and the ground control points needed to map it onto culture polygons in its correct geographic location.

Stage 1 Specific Model Texture. The same as Stage 1 Specific Areal Texture, but with the control points needed to map it onto model polygons.

Stage 2 Specific Areal Texture. A texture file containing an image which has been modified through basic image processing techniques, such as shadow removal and radiometric correction, but has not been modified geometrically. This type of texture includes ground control points.

Stage 2 Specific Model Texture. The same as Stage 2 Specific Areal Texture, but with the control points needed to map it onto model polygons.

Stage 3 Specific Areal Texture. A texture file containing an image which has been orthorectified into a geodetic equal-arc pixel spacing, in addition to being processed as in Stage 2.

Stage 3 Specific Model Texture. The same as Stage 3 Specific Areal Texture, but geometrically mapped into the model coordinate space with equal-distance pixel spacing.

Standard Simulator Data Base (SSDB). The internal information filing structure used by the Simulator Data Base Facility.

Superfeature. An aggregate of individual features within a culture tile into a larger homogeneous data group which identifies all children features belonging to this homogeneous data group.

Terrain Data. As used in this standard, elevation data represented by DTED is considered terrain data.

Texture Data. A two-dimensional raster data set which contains pixel data, usually derived from imagery, which is overlaid on polygonal or other raster data during the real-time rendering process, to increase its spatial frequency.

Tile. Same as Manuscript.

Topology. The establishment of relationships among features in a data set, such that contextual data base queries can be made.

Traversal. The process of identifying the points and edges comprising a feature in a systematic fashion.

VAX/VMS. A proprietary computer operating system developed by the Digital Equipment Corporation, and used by the SDBF.

Vector. Same as Segment.

Vector Product Format. A general-purpose vector data representation standard developed by the Defense Mapping Agency, which will be used as the basis of many of their future cartographic products.

Vertex. Same as Node.

Vertex-to-Vertex Texture. A technique of applying a texture map to a polygon, wherein specific texture pixels are registered to the polygon vertices, and the remainder of the texture pattern is warped to fit the polygon.

Volumetric Modeling. Same as Constructive Solid Geometry.

3.2 Acronyms. For the purpose of this standard, the following acronyms shall apply.

AMSDRL	Acquisition Management Systems and Data Requirements Control List
ANSI	American National Standards Institute
API	Application Programmer's Interface
ASCII	American Standard Code Information Interchange
ASC	Aeronautical Systems Center
BOT	Beginning-of-tape
BPI	Bits Per Inch
BSP	Binary Separating Plane

CDR	Critical Design Review
CSG	Constructive Solid Geometry
DBDD	Data Base Design Document
DBGS	Data Base Generation System
DFAD	Digital Feature Analysis Data
DIAM	Defense Intelligence Agency Manual
DID	Data Item Description
DLMS	Digital Landmass System
DMA	Defense Mapping Agency
DoD	Department of Defense
DP	Distributed Processing
DTED	Digital Terrain Elevation Data
EOF	End of File
EOT	End-of-tape Marker
FACS	Feature Attribute Coding Standard
FID	Feature Identifier
FDC	Feature Descriptor Code
FOM	Figure of Merit
GDS	Gridded Data Section
GFE	Government-Furnished Equipment
GFP	Government-Furnished Property
GTDB	Generic Transformed Data Base
HCV	Hue-Chroma-Value
HDI	High Detail Input/Output
ICMGMS	Interactive Computer Modelling Geometric Modelling System
IGES	Initial Graphics Exchange Specification
JPEG	Joint Photographic Experts Group
JTCG-TSD	Joint Technical Coordinating Group - Training Systems and Devices
LOD	Level of Detail
LSB	Least Significant Bit
LUT	Look-Up Table
MSB	Most Significant Bit
MSL	Mean Sea Level
NIST	National Institute of Standards and Technology
NITF	National Imagery Transmission Format
PDR	Preliminary Design Review
RGB	Red-Green-Blue
SDBF	Simulator Data Base Facility
SIF	SSDB Interchange Format
SIF/DP	SIF for Distributed Processing
SIF/HDI	SIF for High-Detail Input/Output
SMC	Surface Material Category
SSDB	Standard Simulator Data Base
UTM	Universal Transverse Mercator
VMS	Virtual Memory System
VPF	Vector Product Format
WGS	World Geodetic System

4. GENERAL REQUIREMENTS

4.1 External system interface. When invoked as an interface standard, SIF shall be used to transmit and/or receive simulator databases to/from the SDBF or another contractually specified simulator system. Thus, application of this standard may require technical coordination between the sending and the receiving systems. Appendix C gives general guidance on application of this standard.

4.2 Physical medium. SIF was originally designed to be transmitted on sequential-access 9-track magnetic tape. Alternative media may be used upon approval of the acquisition activity (see 6.2).

4.2.1 Physical tape labeling Each SIF tape shall have a physical paper label placed on it consisting of the following information:

SIF Format (always 'SIF/HDI' for SIF/HDI ,
always 'SIF/DP' for SIF/DP)
Transmittal ID
Data Base Title (Short Description)
Volume ID
Originator's Name

4.2.1.1 Transmittal ID. The Transmittal ID is a character string which uniquely identifies the SIF data base exchange. The ID shall be encoded as follows:

YYMMDDOOXX

where YYMMDD = year, month, and day of tape creation,
OO = the originator code, and
XX = sequence number for that day by that originator.

4.2.1.1.1 For example, assume an organization creates a single SIF data base on June 15, 1992. The unique originator's code is '23', and the data base will be sent to two organizations, thus resulting in two transmittals. The transmittal IDs for each set of tape(s) shall be '9206152301' and '9206152302'.

4.2.1.2 Originator codes. Originator codes shall be assigned by the SDBF.

4.2.2 Transmittal form. A transmittal form shall accompany each SIF data base exchange. The SIF Transmittal Form is shown in Figure 1. The information on this form shall include the following:

SIF Format (always 'SIF/HDI' for SIF/HDI ,
always 'SIF/DP' for SIF/DP)
SIF Version Number
Transmittal ID (includes tape creation date)
Data Base Title (Short Description)
Originator's Name & Address
Recipient's Name & Address
Maximum Block Size
Data Types Provided:
- Models
- Culture
- Terrain
- Texture

Number of Tape Volumes
First Volume Contents:
- SIF Data Base Header File Only
- SIF Data Base Header File Plus Data

Volume IDs (in order)

4.3 Quality assurance. The following verification activities shall be conducted to determine the compliance of a candidate data set with this standard. Any data set delivered to the Government with the identification of "SIF" shall have successfully completed all verifications specified herein.

4.3.1 General approach. The quality of SIF data sets shall be assured through a two-pronged validation process, consisting of the formal certification of SIF production processes, as well as the detailed evaluation of selected data sets.

4.3.1.1 Responsibility for test. The producer of a SIF data set shall be responsible for performing all verification testing. Those data sets produced external to the SDBF shall be subjected to formal acceptance testing, as a condition for delivery under the applicable training simulator contracts.

4.3.2 Process certification. The software processes used by external producers of SIF data sets shall be certified as being capable of meeting the data base requirements defined by this standard. Process certification shall consist of three testing activities: format conformance, source correlation, and SSDB compatibility. Based on the performance of the process in these areas, it will be assigned a Figure of Merit (FOM) by the SDBF.

4.3.2.1 Format conformance. This test shall be conducted for both producers and consumers of SIF data sets. Format conformance shall be verified through inspection of the SIF interface and processing software, as well as demonstration of its operation.

4.3.2.1.1 Format conformance producer. Format conformance testing shall be accomplished to verify that the external producer's software is capable of writing data sets which are fully compliant with the format established within this standard.

SIF FORMAT Check The One That Applies: ☐ SIF/HDI MERGED ☐ SIF/HDI LAYERED ☐ SIF/DP	
SIF VERSION NUMBER:	
TRANSMITTAL ID:	
DATA BASE TITLE:	
ORIGINATOR'S NAME:	ORIGINATOR'S ADDRESS:
RECIPIENT'S NAME:	RECIPIENT'S ADDRESS:
MAXIMUM BLOCK SIZE:	
DATA TYPES PROVIDED Check All That Apply: ☐ Models ☐ Culture ☐ Terrain ☐ Texture	
1° x 1° Cells Included/Requested for Culture, Terrain, or Texture Data:	
Non-Referenced Models Included/Requested:	
NUMBER OF TAPE VOLUMES:	
FIRST VOLUME CONTENTS Check The One That Applies: ☐ Data Base Header File Only ☐ Data Base Header File Plus Additional Data	
VOLUME ID'S (IN ORDER):	
RELEASABILITY RESTRICTIONS:	

Figure 1. SIF Transmittal Form.

4.3.2.1.2 Format conformance consumer. Format conformance testing shall be accomplished to verify that a SIF consumer's software is capable of reading SIF data sets which have been generated in compliance with this standard. Format conformance testing shall be performed using Government-supplied SIF test data sets, as defined in paragraph 4.3.4.3, below.

4.3.2.2 Source correlation. This test shall be performed to ensure that a process does not omit or modify information in the conversion to or from the SIF format. This type of certification shall be conducted for both producers and consumers of SIF data sets. This test shall include the inspection of the applicable software code, and an analysis of its algorithms.

4.3.2.2.1 Source correlation producer. For SIF data sets produced externally, it shall be verified that the generation and output process does not eliminate or otherwise corrupt the information contained in the source data base.

4.3.2.2.2 Source correlation consumer. In the case of SIF consumers, it shall be verified that the information processing steps applied to the input SIF data set do not introduce any errors into the data.

4.3.2.3 SSDB compatibility. SSDB compatibility testing shall apply to SIF producers only. This test shall be conducted to verify that the producing DBGS meets the internal quality standards of the SDBF, as defined within this standard and within the SDBF operations concept document, software user's manuals, and standard operating procedures. This test shall be accomplished to ensure that, in addition to being consistent with the information representation schema of the SSDB, the SIF data sets generated by this DBGS are concordant with the information density, level-of-detail allocation, internal linkages, data encoding rules, and other characteristics unique to the SSDB design. This test shall certify that the SIF data set makes the fullest use of standard SIF fields in lieu of User-Defined FACS. This test shall be accomplished through the analysis and demonstration of the producer's DBGS.

4.3.3 Data set verification. Verification of SIF data sets shall be conducted by their producers, with documentation provided to the Government as evidence of their having been successfully verified. The scope of the product verification activity shall be dependent upon the amount of process verification performed, the size and criticality of the data bases generated, and other factors, as jointly determined by the procuring activity and the SDBF.

4.3.3.1 Verification of SIF product. Representative SIF data sets shall be tested for compliance with this standard. Analogous to the process certification testing described above, data sets shall be tested for format conformance, source correlation, and SSDB compatibility. Product verification shall initially be performed in conjunction with the certification of the producer DBGS. Subsequent product verification testing may be performed on select data sets, in accordance with the terms of the applicable contract. Each data set shall identify the Figure of Merit assigned to its producing DBGS by the SDBF.

4.3.3.1.1 Format conformance. This test shall be used to ensure that the information contained in the data set meets the defined format specification. Format compliance shall be assured through inspection. When furnished by the Government, the SIF producer shall use an automated format verification tool to aid in this task.

4.3.3.1.2 Source correlation. This test shall be used to verify that the information contained within the SIF data set matches that contained within the original data base from which it was derived. Source correlation compliance shall be assured through inspection and analysis. Graphical displays and/or plots shall be generated for both source and SIF data bases, to facilitate a side-by-side visual comparison of data base contents. Statistics shall be generated to provide a comparison of the contents of the two data sets.

4.3.3.1.3 SSDB compatibility. As a minimum, SSDB compatibility shall be verified through analysis and inspection of the product data set. At the discretion of the SDBF, compatibility may be further verified through the actual processing of the data set through the SDBF software, resulting in its successful integration into the SSDB.

4.3.3.2 Verification of SIF application. When a SIF data base is provided to the consumer by the Government, it shall be verified that the contractor has the ability to correctly interpret and utilize the information contained therein. Application compliance shall be tested in two ways: accommodation and utilization.

4.3.3.2.1 Accommodation. This test shall be used to verify that the contractor's hardware and software is capable of reading the SIF media and interpreting its contents correctly. SIF accommodation shall be verified through demonstration, using a test SIF data base furnished by the Government.

4.3.3.2.2 Utilization. Utilization testing shall ensure that the consumer's data base(s) incorporate the information contained in the SIF data set. It shall be verified that the information content of the Government-provided SIF data set is reflected in the contractor-generated trainer data base, as well as any intermediate data base(s) used by an external DBGS. This testing shall be accomplished by means of inspection and demonstration.

4.3.4 Tools and test data. An Application Programmer's Interface (API) toolkit has been developed for SIF users. The toolkit consists of a top-level main program module, SIF data structure module, SIF read/write modules, SIF validation module, coordinate transformation module, color conversion module, query SIF module, browser module, and terrain grid orientation module. The API toolkit is in 'C' with a user and source manual and can be obtained GFE through the SDBF.

4.3.4.1 Government-furnished tools. When furnished as GFE, the contractor shall use any SIF validation tools developed under Project 2851 or by the SDBF.

4.3.4.2 Contractor-developed tools. Subject to Government approval, contractor-developed software tools may be used as verification aids.

4.3.4.3 Government-furnished test data sets. The SDBF will provide test data sets in support of verification testing. SIF consumer programs shall use these data sets in demonstrating compliance with the SIF standard, as specified above.

4.3.4.4 Contractor-developed test data sets. The contractor shall develop additional test data sets as required to demonstrate SIF compatibility.

4.3.5 Test documentation. SIF verification testing shall be addressed within the system test plan. SIF test procedures shall be developed and performed in accordance with this plan. Test reports shall be delivered to the Government, documenting the results of the above verifications.

4.3.6 Exceptions. In certain instances, an external SIF producer may be unable to fully meet the acceptability criteria specified herein. In such a case, a petition may be submitted to the cognizant acquisition agency, requesting that a waiver be granted, so that the data set may be delivered in fulfillment of the SIF requirement. The petition shall include a detailed analysis illustrating why the requirement cannot be met. It shall also provide evidence that the data set is of sufficient value that its inability to meet these criteria are exceeded by the benefits of its inclusion in the SSDB. This petition will be evaluated by the SDBF, which will advise the acquisition agency on whether or not it is in the Government's best interest to grant the waiver.

4.4 Documentation. Each SIF data set shall be documented (see 6.5). In general, the data shall include a description of those characteristics which make that particular SIF data set unique. Specific information which shall be contained in each SIF data set is as follows.

4.4.1 Application. The SIF data set shall provide a description of application for which the data base was originally intended. The document shall identify those aspects of the data base which have been specifically tailored for the purposes of this application.

4.4.2 Training utility. The training utility of an externally generated SIF data set shall be described in sufficient detail to allow the SDBF to make a determination as to its value as a component of the SSDB.

4.4.3 Content. The SIF data set shall delineate the specific content in terms of the areas of coverage, sources used, areas of high detail, models included, texture types, and other relevant information. The SIF data set shall include illustrative plots, drawings, graphs, and tables as required to describe the contents of the data set.

4.4.4 Indigenous standards. The SIF data set shall identify any indigenous standards and procedures used in the creation of the data. These shall be documented to the extent that they vary from the internal standards of the SDBF, or that they clarify ambiguous aspects of the SDBF standards.

4.4.5 Transformation. The SIF data set shall describe the transformation processes used in converting the source data base into SIF/HDI, and the quality assurance tests performed on the converted data set.

4.4.6 Utilization instructions. Instructions for the assembly of dissimilar data types, such as the integration of model complexes into their underlying terrain, feature, and texture data environment, shall be included in the SIF data set.

5. DETAILED REQUIREMENTS

5.1 Standard Simulator Data Base Interchange Format (SIF)/High Detail Input/Output (HDI) data base

5.1.1 SIF/HDI data base structure

5.1.1.1. Logical format. The logical format of a SIF/HDI data base is made up of a hierarchy of data entities. The hierarchy is as follows:

Data Base

  Section

    File

      Record

        Field

          Item

5.1.1.1.1 Data base. The data base shall consist of a data base header file and all the requested models, culture, terrain, and texture for a specified geographic area. Models that are not located in the geographic area can also be explicitly requested. Logically, the data base consists of a data base header file and one, two, or three sections.

5.1.1.1.2 Section. A section is a series of files consisting of information for a certain type of data: (1) models, (2) culture, or (3) terrain and texture. Within a database, there is either one section or no sections for each of these three types.

5.1.1.1.3 File/record/field/item. A section is made up of a series of files, a file consists of a series of records, a record consists of a series of fields, and a field consists of one or more items. The item is the lowest logical data entity defined within SIF.

5.1.1.2 Physical format. The physical format of the SIF/HDI data base shall be as described in the following sections in terms of the data order, the physical tape format, and the general file and data formats.

5.1.1.2.1 Data order. The physical order of data in the SIF/HDI data base shall be as follows:

SIF/HDI Data Base Header File
Model Data Section [optional]
Culture Data Section [optional]
Gridded (Terrain and Texture) Data Section [optional]

5.1.1.2.1.1 The first file on the first tape of the data base shall be the SIF/HDI Data Base Header File. It contains control information, including counts of various data entities as well as the file name of each file in the data base. The order in which the file names appear in the SIF/HDI Data Base Header File is the order in which those files shall appear on the tape(s).

5.1.1.2.2 Physical tape format. The physical tape format of a SIF/HDI data base shall be Level 3 of the ANSI Standard for Magnetic Tape Labels and File Structure for Information Interchange, ANSI X3.27-1978. The format of the physical tape shall be as follows:

Beginning-of-Tape Marker (BOT)

Volume Label (VOL1)

for each file
- File Header Labels (HDR1, HDR2)
- Tape Mark (TM)
- File Section
- Tape Mark (TM)
- File Trailer Labels (EOF1, EOF2 or EOV1, EOV2)

Tape Mark (TM)

Tape Mark (TM)

Scratch Tape

End-of-Tape Marker (EOT)

5.1.1.2.2.1 Four file/volume configurations shall be supported. They are single file/single volume; single file/multi-volume; multi-file/single volume; and multi-file/multi-volume. A SIF/HDI data base may span several tape volumes. An individual file may cross a tape boundary; in such a case, EOV1 and EOV2 tape labels shall be written after an EOT and a tape mark at the end of the tape. When a file ends within a tape, it shall be followed by a tape mark and then the file trailer labels EOF1 and EOF2.

5.1.1.2.2.2 Tapes shall be written at 6250 bits per inch (bpi) with the GCR recording method. The block length shall be denoted by the Block Length Field within a file's HDR2 label. Block size can vary from file to file. The allowed minimum tape block size shall be 2048 bytes while the maximum shall be 65534 bytes.

5.1.1.2.2.3 Only the characters A through Z, 0 through 9, and the special characters '&', '-', '_', and '$' shall be used in filenames. The period may appear once within the name with a maximum of three characters following it. The file name shall have no more than 17 characters.

5.1.1.2.3 General file and data formats. The file and data formats are detailed for the SIF/HDI Data Base Header File and each of the data sections in section 5.1.2 of this document.

5.1.1.2.3.1 Non-gridded data files. The SIF/HDI Data Base Header File and all files in the Model Data and Culture Data sections, except where explicitly noted otherwise, shall be in a compressed ASCII format with record keyword separators and ASCII null ('00') field separators. Within any of these files, when a field is initially all blanks, it shall be compressed to a null field of zero length; thus, two consecutive field separators shall occur at this point. There shall be one or more ASCII CNTRL-Z characters at the end of each ASCII file.

5.1.1.2.3.2 Gridded data files. The Gridded Data section, containing both terrain elevation and rasterized texture data, shall have its files stored in the specified NITF format. All header files shall be stored in non-compressed ASCII, while the data files containing the actual grid data shall be in a binary format as specified by the NITF standard.

5.1.1.2.3.3 Non-ASCII files. Non-ASCII files shall be in a binary format where integer data are stored in two's-complement, with the high-order bit in the high-order byte representing the sign, as shown in Figure 2. Floating point data are stored in a single-precision format, as defined by ANSI/IEEE Std 754, Binary Floating Point Arithmetic. Appendix A shows the number of bytes used for each data field.

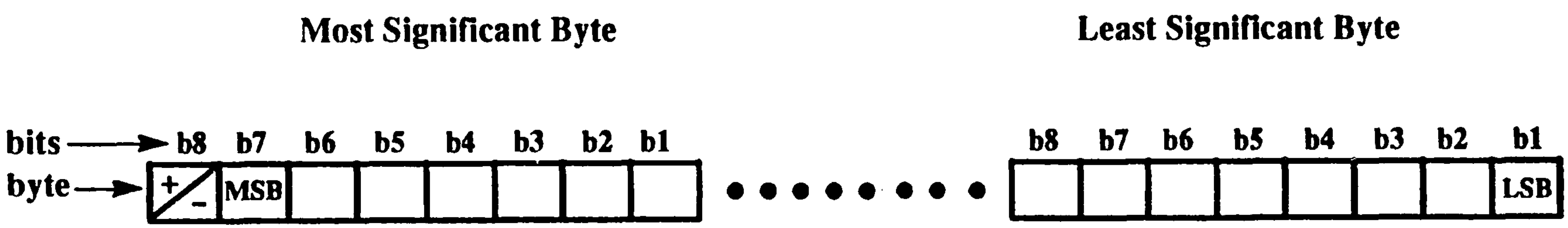
5.1.2 SIF/HDI file formats

5.1.2.1 SIF/HDI Data Base Header File Format. The SIF/HDI Data Base Header Format shall consist of a single file that contains general transmittal, identification, and directory information.

5.1.2.1.1 Header data encoding. A compressed form of ASCII shall be used in this file. The compression shall consist of stripping all leading zeros and blanks from numeric strings and all leading and trailing blanks from character strings. Every ASCII field shall be a variable-length field, separated by the ASCII null character ('00'). Since fields are variable-length, records shall also vary in length. Every record (except the file identifier record) begins with a 2-character keyword identifying its type. The record keyword for a comment record is identified as consecutive asterisks (**). Following the keyword is the standard ASCII null character ('00') as the field separator. The comment field will then continue until end of file (EOF) or the end of field separator ('00') is located in the SIF data file. Comment records shall not occur in the middle of any record in the file, but can be placed before or after any other record.

5.1.2.1.2 Header section structure. The Header section shall consist of a single SIF/HDI Data Base Header File.

5.1.2.1.3 Header file structure. This mandatory file shall contain general transmittal, identification, and directory information concerning the SIF/HDI data base to follow. It shall be the first file on the first tape volume. The order of data in the SIF/HDI Data Base Header File is as specified below. The order in which the file names appear in this file is the required order in which the files shall appear in the data base.



MSB = Most Significant Byte

LSB = Least Significant Byte

Figure 2. SIF/HDI Binary Integer Format.

a. The file name of the SIF/HDI Data Base Header shall be:

"SIFHDI.HDR".

b. The SIF/HDI Data Base Header file format shall be as follows:

```
SIF File Identifier Record
Transmittal Description Record
Data Directory Record

2D Static Model Library Header File Name Record
for each 2D static model
   2D Static Model Entry Record

3D Static Model Library Header File Name Record
for each 3D static model
   3D Static Model Entry Record

3D Dynamic Model Library Header File Name Record
for each 3D dynamic model
   3D Dynamic Model Entry Record

Model Table File Name Record

Culture Header File Name Record

for each culture tile
   Culture Tile Entry Record

NITF Header File Name Record

for each terrain tile
   Terrain Tile Entry Record

for each generic texture
   Generic Texture Entry Record

for each stage 3 specific model texture
   Stage 3 Specific Model Texture Entry Record

for each stage 2 specific model texture
   Stage 2 Specific Model Texture Entry Record

for each stage 1 specific model texture
   Stage 1 Specific Model Texture Entry Record

for each stage 3 specific areal texture
   Stage 3 Specific Areal Texture Entry Record

for each stage 2 specific areal texture
   Stage 2 Specific Areal Texture Entry Record

for each stage 1 specific areal texture
   Stage 1 Specific Areal Texture Entry Record

for each SMC/FDC texture
   SMC/FDC Texture Entry Record
```

5.1.2.1.3.1 SIF File Identifier Record. The field structure of this record shall be as follows:

File Identifier Field (always 'SIF/HDI DATA BASE HEADER')

5.1.2.1.3.2 Transmittal Description Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'TD')
SIF Format Field
Originator Field
Recipient Field
Transmittal ID Field
Creation Date Field
Source Agency/Project Field
Database Name Field
Data On This Volume Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field
SIF Version Number Field

5.1.2.1.3.3 Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'DD')
Number of 2D Static Models Field
Number of 3D Static Models Field
Number of 3D Dynamic Models Field
Number of Culture Tiles Field
Number of Terrain Tiles Field
Number of Generic Textures Field
Number of Stage 3 Specific Model Textures Field
Number of Stage 2 Specific Model Textures Field
Number of Stage 1 Specific Model Textures Field
Number of Stage 3 Specific Areal Textures Field
Number of Stage 2 Specific Areal Textures Field
Number of Stage 1 Specific Areal Textures Field
Number of SMC/FDC Textures Field
Merged or Layered Culture Field
Data Base SW Corner Field
Data Base NE Corner Field

5.1.2.1.3.4 Two-dimensional (2D) Static Model Library Header File Name Record. This record shall be included when the number of 2D Static Models Field in the Data Directory Record is non-zero. The field structure of this record shall be as follows:

Record Keyword Field (always '2L')
File Name Field

5.1.2.1.3.5 <u>2D Static Model Entry Record</u>. The number of these records shall correspond to the number of 2D Static Models Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always '2S')
Model Data File Name Field
Vertex Table File Name Field
Model Number Field
Model Name Field
Model Description Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.1.2.1.3.6 <u>Three-dimensional (3D) Static Model Library Header File Name Record</u>. This record shall be included when the number of 3D Static Models Field in the Data Directory Record is non-zero. The field structure of this record shall be as follows:

Record Keyword Field (always '3L')
File Name Field

5.1.2.1.3.7 <u>3D Static Model Entry Record</u>. The number of these records shall correspond to the Number of 3D Static Models Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always '3S')
Model Data File Name Field
Vertex Table File Name Field
Model Number Field
Model Name Field
Model Description Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.1.2.1.3.8 <u>3D Dynamic Model Library Header File Name Record</u>. This record shall be included when the number of 3D Dynamic Models Field in the Data Directory Record is non-zero. The field structure of this record shall be as follows:

Record Keyword Field (always 'DL')
File Name Field

5.1.2.1.3.9 3D Dynamic Model Entry Record. The number of these records shall correspond to the number of 3D Dynamic Models Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always '3D')
Model Data File Name Field
Vertex Table File Name Field
Model Number Field
Model Name Field
Model Description Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.1.2.1.3.10 Model Table File Name Record. This record shall be included if any of the number of models fields in the Data Directory Record is non-zero. If any table file listed herein does not exist, then the file name is represented by the null field. (The null field is indicated by consecutive null '00' separators.) The field structure of this record shall be as follows:

Record Keyword Field (always 'MT')
Data Source Table File Name Field
FACS Table File Name Field
User-Defined FACS Table File Name Field
Color Table File Name Field
Face-Based Texture Reference Table File Name Field
Vertex-to-Vertex Texture Reference Table File Name Field
Model-Based Texture Reference Table File Name Field
Non-Mapped Texture Reference Table File Name Field

5.1.2.1.3.11 Culture Header File Name Record. This record shall be included when the number of Culture Tiles Field in the Data Directory Record is non-zero. If a file does not exist, then the file name is represented by the null field. The field structure of this record shall be as follows:

Record Keyword Field (always 'CH')
Data Base Header File Name Field
Tile Information File Name Field

5.1.2.1.3.12 <u>Culture Tile Entry Record</u>. The number of these records shall correspond to the number of Culture Tiles Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'CE')
Two-D Coordinate File Name Field
Three-D Coordinate File Name Field
FACS Table File Name Field
User-Defined FACS Table File Name Field
Color Table File Name Field
FID/FDC Reference Table File Name Field
Global-Based Texture Reference Table File Name Field
Non-Mapped Texture Reference Table File Name Field
Model Reference Table File Name Field
Superfeature File Name Field
Feature File Name Field
Segment File Name Field
SW Corner Field
NE Corner Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.1.2.1.3.13 <u>NITF Header File Name Record</u>. This record shall be included when the number of Terrain Tiles Field and/or any of the number of Textures Fields is non-zero. The field structure of this record shall be as follows:

Record Keyword Field (always 'NH')
File Name Field

5.1.2.1.3.14 <u>Terrain Tile Entry Record</u>. The number of these records shall correspond to the Number of Terrain Tiles Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'TE')
Terrain Sub-Header File Name Field
Terrain Data File Name Field
Horizontal Resolution Field
Vertical Resolution Field
SW Corner Field
NE Corner Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.1.2.1.3.15 Generic Texture Entry Record. The number of these records shall correspond to the Number of Generic Textures Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'GX')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Creation Date and Time Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.1.2.1.3.16 Stage 3 Specific Model Texture Entry Record. The number of these records shall correspond to the number of Stage 3 Specific Model Textures Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'M3')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Capture Date and Time Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.1.2.1.3.17 <u>Stage 2 Specific Model Texture Entry Record</u>. The number of these records shall correspond to the number of Stage 2 Specific Model Textures Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'M2')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Capture Date and Time Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.1.2.1.3.18 <u>Stage 1 Specific Model Texture Entry Record</u>. The number of these records shall correspond to the number of Stage 1 Specific Model Textures Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'M1')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Capture Date and Time Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.1.2.1.3.19 Stage 3 Specific Areal Texture Entry Record. The number of these records shall correspond to the number of Stage 3 Specific Areal Textures Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'A3')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Capture Date and Time Field
SW Corner Field
NE Corner Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.1.2.1.3.20 Stage 2 Specific Areal Texture Entry Record. The number of these records shall correspond to the number of Stage 2 Specific Areal Textures Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'A2')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Capture Date and Time Field
SW Corner Field
NE Corner Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.1.2.1.3.21 Stage 1 Specific Areal Texture Entry Record. The number of these records shall correspond to the number of Stage 1 Specific Areal Textures Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'A1')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Capture Date and Time Field
SW Corner Field
NE Corner Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.1.2.1.3.22 SMC/FDC Texture Entry Record. The number of these records shall correspond to the number of SMC/FDC Textures Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'SF')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Capture Date and Time Field
SW Corner Field
NE Corner Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.1.2.2 Model data

5.1.2.2.1 Model data encoding. A compressed form of ASCII shall be used. The compression shall take the form of stripping all leading zeros and blanks from numeric strings and all leading and trailing blanks from character strings. Every ASCII field shall be a variable-length field, separated by the ASCII null character ('00'). Since fields are variable-length, records shall also vary in length. Every record (except the SIF file identifier record) shall begin with a 2-character keyword identifying its type. The record keyword for a comment record shall be two consecutive asterisks (**). Following the keyword is the standard ASCII null character ('00') as the field separator. The comment field shall then continue until end of file (EOF) or the end of field separator ('00') is located in the SIF data file. Comment records shall not occur in the middle of any record in the file, but can be placed before or after any other record in the data file. Items in a field are separated by 'space' characters.

5.1.2.2.1.1 Model building standards. Models shall be constructed using a right-handed X-Y-Z Cartesian coordinate system. Models shall be built with the local X-axis identifying the direction of the front of the model, and the Z-axis pointing straight up into the air. For a static model, the front shall be defined as the side facing the nearest road feature. For a dynamic model, the X-axis shall point in the normal direction of motion; however, any dynamic model that launches vertically shall be modeled with its Z-axis pointing vertically. The origin of a static model shall be defined as a point where the model touches the earth. If the model is to appear floating over the earth, it shall have its origin at the point directly below it on the earth. The origin shall be at the center of the base of the model in the X-Y plane. For dynamic models, in the X-Y plane, the origin shall be the centroid of the model. The elevation of the origin shall be where the wheels, tracks, skids, or pontoons contact the ground if it is a surface vehicle, aircraft, or helicopter.

5.1.2.2.2 Model section structure. Within a SIF data base, models shall be organized into three general classes: 2-D static models, 3-D static models, and 3-D dynamic models. Each type shall have a single library header file which shall in turn refer to separate Model Files containing the actual model representations. The SIF data base shall support storage of each model at up to nine levels of detail (LODs). LOD 0 shall have the least amount of detail, while LOD 8 has the most detail. A series of tables shall be used to refer to colors, face-based texture references, vertex-to-vertex texture references, model-based texture references, user-defined FACS, and the SIF-defined FACS. Each SIF model shall be described by a file made up of variable-length logical keyword records containing ASCII alphanumeric strings. This file shall consist of both geometry and attribute information. If polygonal geometry exists, then a binary vertex table file shall exist to describe polygon vertices. All models shall share the auxiliary data found in the table files. The IGES Version 4.0 file format shall be used to describe the constructive solid geometry of a model. The SIF/HDI format for models shall be entirely ASCII.

5.1.2.2.2.1 Field format. Data fields and records shall vary in length. They shall be stored in a compressed form of ASCII unless otherwise noted in this standard. (The Vertex Table File shall be stored in binary format.) All records (except the file identifier record and table entry records) shall begin with a 2-character keyword identifier. Items in a field are separated by 'space' characters.

5.1.2.2.2.2 Section format. The SIF/HDI model section format shall be as follows and as shown in Figure 3.

For each model library type
    Model Library Header File
    For each model
        Model Data File
        Vertex Table File [mandatory for
          polygonal format only]
Data Source Table File
FACS Table File [optional]
User-Defined FACS Table File [optional]
Color Table File [optional]
Face-Based Texture Reference Table File [optional]
Vertex-to-Vertex Texture Reference Table File [optional]
Model-Based Texture Reference Table File [optional]
Non-Mapped Texture Reference Table File [optional]

5.1.2.2.3 Model file structures

5.1.2.2.3.1 Model Library Header File. There shall be a separate Model Library Header File for each of the three library types. These files shall be named "MODEL2DS.LHD" for the 2D Static Model Library, "MODEL3DS.LHD" for the 3D Static Model Library, and "MODEL3DD.LHD" for the 3D Dynamic Model Library. The Model Library Header File format shall be as follows:

SIF File Identifier Record
Model Library Header Record

5.1.2.2.3.1.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI MODELS')
File Identifier Field (always 'MODEL LIBRARY HEADER')

5.1.2.2.3.1.2 Model Library Header Record. The field structure of this file shall be as follows:

Record Keyword Field (always 'ML')
Model Library Type Field
Security Level Field
Number of Models Field

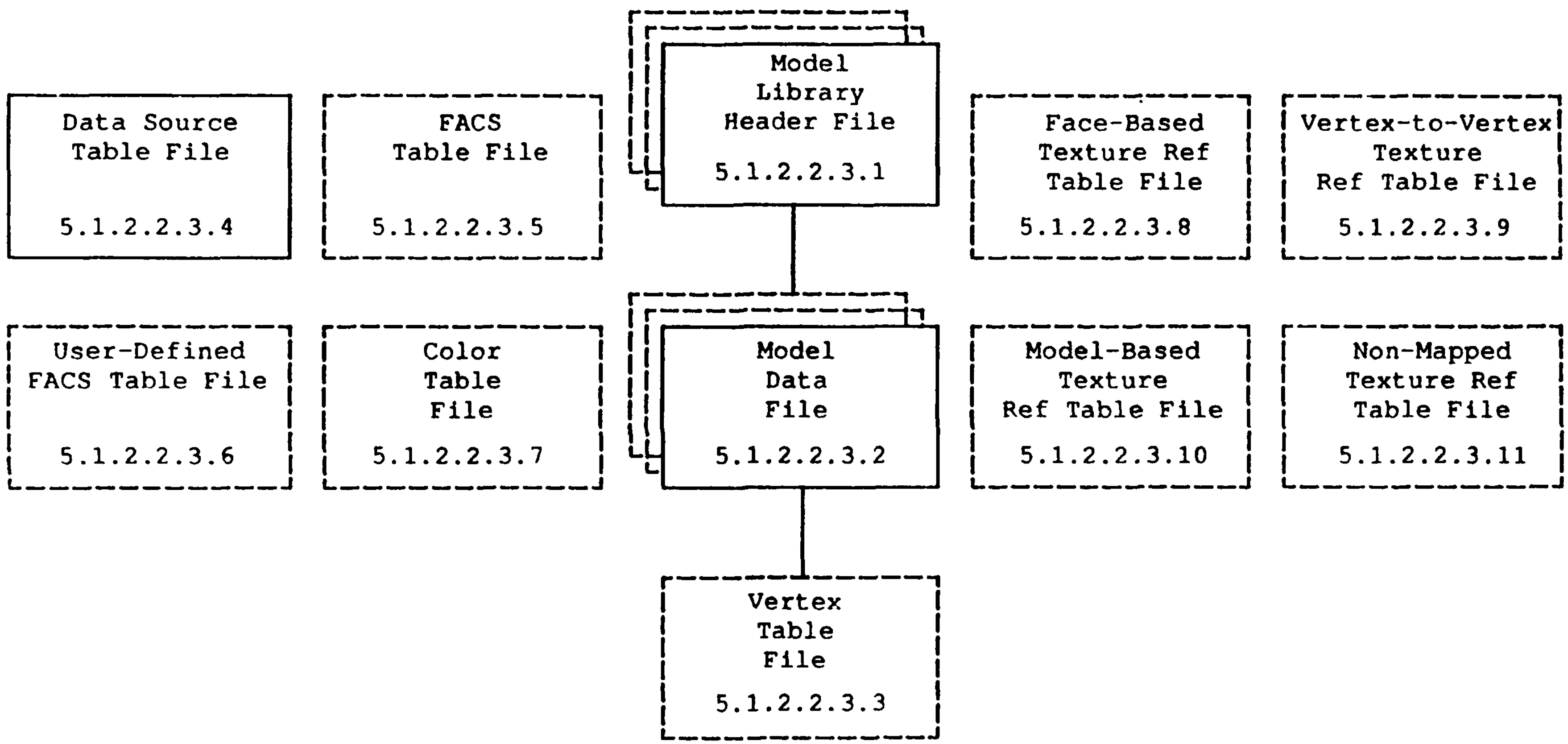


Figure 3. SIF/HDI Model Data File Relationships.

5.1.2.2.3.2 Model Data File. There shall be one Model Data File for each model in the data base. File names shall be of the form "Mtttxxxxx.DAT", where "ttt" is "2DS" for a 2D Static Model, "3DS" for a 3D Static Model, "3DD" for a 3D Dynamic Model, and "xxxxx" is the model sequence number (not the SSDB model number). The Model Data File format shall be as follows and as shown in Figure 4.

```
SIF File Identifier Record
Model Header Record
For each LOD
   LOD Header Record
   Cluster Statistics Record(s) [optional]
   Separation Plane Record(s) [optional]
   Subsidiary Model Reference Record(s) [optional]
   Point Light String Record(s) [optional]
   if MODEL_FORM = POLYGONAL_ONLY or CSG_AND_POLYGONAL then
      Collision Test Point Record(s) [optional]
      Model LOD Texture Reference Pointer Record(s) [optional]
   end if
   if MODEL_FORM = CSG_ONLY or CSG_AND_POLYGONAL then
      IGES Start Record
      IGES Records
      Polygonizing Instruction Records
   end if
   for each component
      Component Header Record
      Microdescriptor Record(s) [optional]
      if MODEL_FORM = POLYGONAL_ONLY or CSG_AND_POLYGONAL then
              Component Texture Reference Pointer Record(s) [optional]
           for each polygon
              Polygon Header Record
              Vertex Pointer Record(s)
              Vertex Normal Record(s) [optional]
              Vertex Color Record(s) [optional]
              Polygon Microdescriptor Record(s) [optional]
              Polygon Texture Reference Pointer Record(s) [optional]
            end if
```

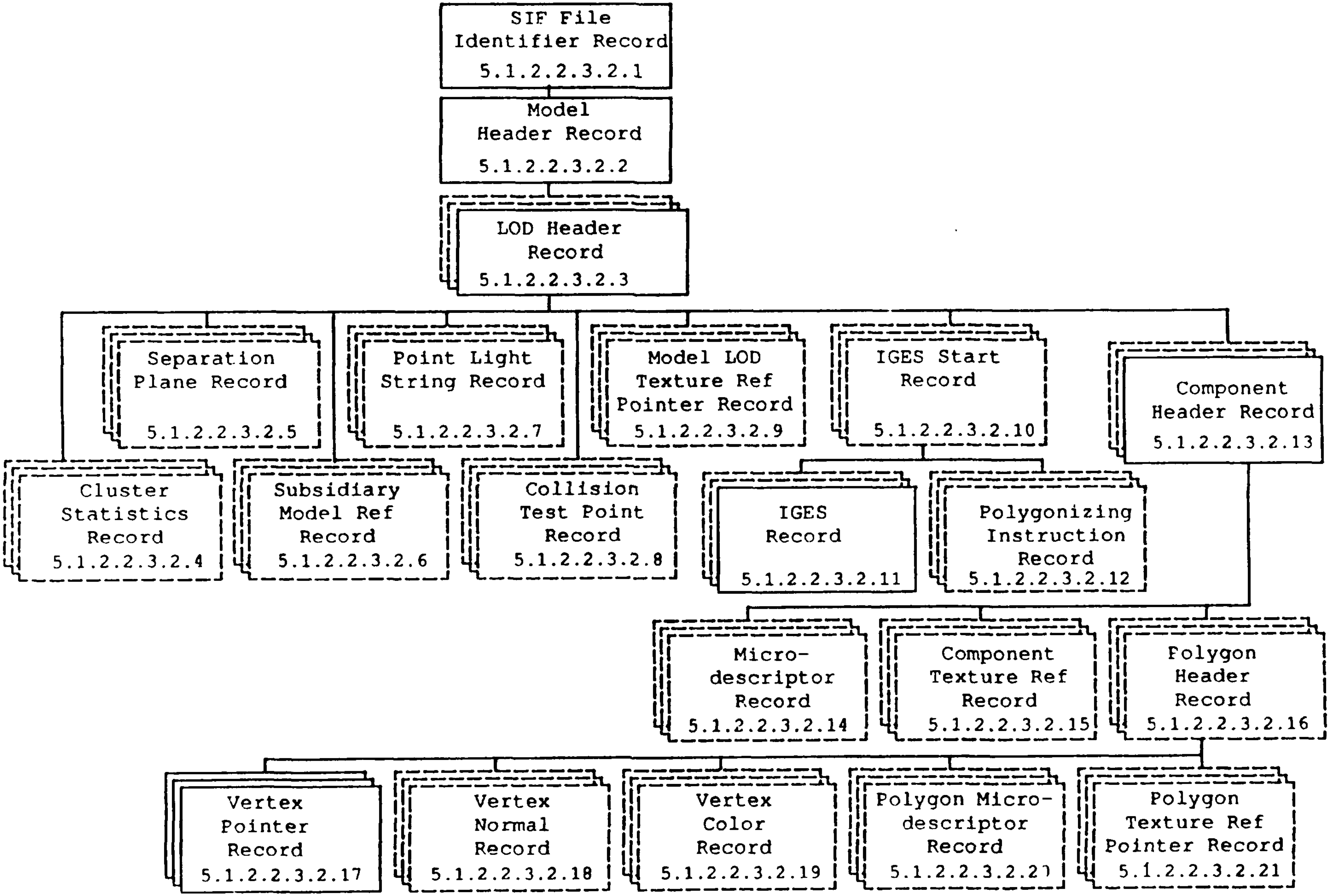


Figure 4. SIF/HDI Model Data File's Record Relationships.

5.1.2.2.3.2.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI MODELS')
File Identifier Field (always 'MODEL DATA')

5.1.2.2.3.2.2 Model Header Record. The number of these records within a Model Data File shall be one. The field structure of this record shall be as follows:

Record Keyword Field (always 'MH')
Model Number Field
Model Name Field
Model Form Field
Model Description Field
Security Level Field
Model Library Type Field
Sensors Supported Field
Source Simulator Field
Last Maintenance Date Field
Number of Model LODs Field
Number of Model Vertices Field
Generic Model Flag Field
Feature Descriptor Code Field
AV Code 1 Field
AV Code 2 Field
AV Code 3 Field
FACS Table Index Field (defaults to 0 if no optional fields specified)
Number of Data Sources Field
Data Source Table Pointer List Subrecord

5.1.2.2.3.2.2.1 Data Source Table Pointer List Subrecord. The field structure of this subrecord shall be as follows:

for each data source
  Data Source Table Index Field

5.1.2.2.3.2.3 LOD Header Record. The number of these records for a given model group shall correspond to the value contained in the Number of Model LODs field in the Model Header record. The field structure of this record shall be as follows:

```
Record Keyword Field  (always 'LH')
Model LOD Field
LOD Resolution Description Field
Number of Components Field
Number of Polygons Field
Number of Edges Field
Number of Vertices Field
Number of Subsidiary Model References Field
Number of Clusters Field
Number of Separation Planes Field
All Convex Clusters Flag Field
P2851 Binary Separation Planes Flag Field
Number of Point Light Strings Field
if MODEL_FORM = POLYGONAL_ONLY or CSG_AND_POLYGONAL then
   All Convex Polygons Flag Field
   Number of Collision Test Points Field
   Number of Model LOD Texture References Field
end if
FACS Table Index Field (defaults to 0 if
                        no optional fields specified)
```

5.1.2.2.3.2.4 Model Cluster Statistics Record. The number of cluster statistics records shall correspond to the value in the Number of Clusters Field in the LOD Header Record. The field structure of this record shall be as follows:

```
Record Keyword Field  (always 'CS')
Cluster ID Field
Convex Cluster Flag Field
Number of Polygons Field
Number of Edges Field
Number of Vertices Field
FACS Table Index Field (defaults to 0 if
                        no optional fields specified)
```

5.1.2.2.3.2.5 Separation Plane Record. The number of these records shall correspond to the Number of Separation Planes field in the parent LOD Header record. The field structure of this record shall be as follows:

```
Record Keyword Field (always 'SP')
if MODEL_FORM = POLYGONAL_ONLY or CSG_AND_POLYGONAL then
   Polygon ID Field
end if
Separation Plane Number Field
Separation Plane Coefficients Field
```

5.1.2.2.3.2.6 Subsidiary Model Reference Record. The number of these records for a given model shall correspond to the value contained in the Number of Subsidiary Model References field in the parent LOD Header record. The field structure of this record shall be as follows:

Record Keyword Field (always 'MR')
Referenced Model Library Type Field
Referenced Model Number Field
Referenced Model LOD Field
Translation Field
Scale Factor Field
Rotation Angles Field
Articulated Part Flag Field
FACS Table Index Field

5.1.2.2.3.2.7 Point Light String Record. The number of Point Light String records will correspond to the value in the Number of Point Light Strings field within the LOD Header record. The field structure of this record shall be as follows:

Record Keyword Field (always 'LS')
Length Field
Orientation Field
Shape Code Field
Width Field
Directionality Field
Light Type Field
Source ID Number Field
Predominant Height Field
Surface Material Category Field
Color Table Index Field
Layer Number Field
Number of Lights Field
Point Light Positions Subrecord
FACS Table Index Field (defaults to 0 if no optional fields specified)

5.1.2.2.3.2.7.1 Point Light Positions Subrecord. The field structure shall be as follows:

for each light in the string
Point Light Position Field

5.1.2.2.3.2.8 Collision Test Point Record. The number of these records shall correspond to the value in the Number of Collision Test Points field within the parent LOD Header record. The field structure of each record shall be as follows:

Record Keyword Field (always 'TP')
Vertex List Position Field

5.1.2.2.3.2.9 Model LOD Texture Reference Pointer Record. The number of these records shall correspond to the value in the Number of Model LOD Texture References field within the parent LOD Header record. The field structure of each record shall be as follows:

Record Keyword Field (always 'LR')
Texture Mapping Type Field
Texture Reference Table Index Field
Texture Mapping Set ID Field

5.1.2.2.3.2.10 Initial Graphics Exchange Specification (IGES) Start Record. If the Model Form is CSG only, or both CSG and polygonal, then there shall be exactly one of these records. The field structure of this record shall be as follows:

Record Keyword Field (always 'IS')
Number of Polygonization Instructions Field
Number of IGES Records Field

5.1.2.2.3.2.11 IGES Records. These records shall be of the form specified in the IGES Standard, Version 4.0.

5.1.2.2.3.2.12 Polygonization Instruction Record. The number of these records shall correspond to the value in the Number of Polygonization Instructions field within the IGES Start record. The field structure of each record shall be as follows:

Record Keyword Field (always 'PI')
Sequence Number Field
Number of Polygons Along Surface 1 Field
Number of Polygons along Surface 2 Field (as required)

5.1.2.2.3.2.13 Component Header Record. The number of these records shall correspond to the Number of Components field in the LOD Header Record. The field structure of this record shall be as follows:

```
Record Keyword Field  (always 'CH')
Component ID Field
if MODEL_FORM = CSG_ONLY or CSG_AND_POLYGONAL then
   IGES Sequence Number for Component Field
end if
Color Table Index Field
if MODEL_FORM = POLYGONAL_ONLY or CSG_AND_POLYGONAL then
   Number of Polygons Field
   Number of Component Texture References Field
end if
Number of Microdescriptors Field
FACS Table Index Field (defaults to 0 if
                        no optional fields specified)
```

5.1.2.2.3.2.14 Model Microdescriptor Record. The number of these records corresponds to the Number of Microdescriptors field in the Component Header Record. These microdescriptors apply to all faces (polygons) in the component. The field structure of this record shall be as follows:

Record Keyword Field (always 'MI')
Microdescriptor Type Field
Microdescriptor Value Field

5.1.2.2.3.2.15 Component Texture Reference Pointer Record. The number of these records shall correspond to the value in the Number of Component Texture References field within the parent Component Header Record. The field structure of each record shall be as follows:

Record Keyword Field (always 'CR')
Texture Mapping Type Field
Texture Reference Table Index Field
Texture Mapping Set ID Field

5.1.2.2.3.2.16 Polygon Header Record. The number of these records for a given model shall correspond to the value contained in the Number of Polygons field in the parent LOD Header record. The number of these records for a given component shall correspond to the value contained in the Number of Polygons field in the parent Component Header record. The field structure of this record shall be as follows:

Record Keyword Field (always 'PO')
Polygon ID Field
Component ID Field
Cluster ID Field
Convex Polygon Flag Field
Number of Microdescriptors Field
Number of Vertices Field
Number of Vertex Normals Field
Number of Vertex Colors Field
Number of Polygon Texture References Field
FACS Table Index Field (defaults to 0 if no optional fields specified)

5.1.2.2.3.2.17 Vertex Pointer Record. The number of these records shall correspond to the Number of Vertices field within the parent Polygon Header record. The field structure of each record shall be as follows:

Record Keyword Field (always 'VP')
Vertex List Position Field

5.1.2.2.3.2.18 Vertex Normal Record. The number of these records shall correspond to the Number of Vertex Normals field within the parent Polygon Header record. This number shall either be zero or the same as the Number of Vertices field. The Normal List and the Vertex List used by the Vertex Pointer Record shall be combined into the same vertex file. The field structure of each record shall be as follows:

Record Keyword Field (always 'VN')
Normal List Position Field

5.1.2.2.3.2.19 Vertex Color Record. The number of these records shall correspond to the Number of Vertex Colors field within the parent Polygon Header record. This number shall either be zero or the same as the Number of Vertices field. The order of the vertex colors shall follow the same order as the vertex pointers for the current polygon. The field structure of each record shall be as follows:

Record Keyword Field (always 'VC')
Color Table Index Field

5.1.2.2.3.2.20 Polygon Microdescriptor Record. The number of these records shall correspond to the Number of Microdescriptors field in the Polygon Header Record. These microdescriptors shall override those of the parent component. The field structure of this record shall be as follows:

Record Keyword Field (always 'PM')
Microdescriptor Type Field
Microdescriptor Value Field

5.1.2.2.3.2.21 Polygon Texture Reference Pointer Record. The number of these records shall correspond to the value in the Number of Polygon Texture References field within the parent Polygon Header record. The field structure of each record shall be as follows:

Record Keyword Field (always 'PR')
Texture Mapping Type Field
Texture Reference Table Index Field
Texture Mapping Set ID Field

5.1.2.2.3.3 Vertex Table File. The name of this file shall be of the form "Mtttxxxxx.VTX", where "ttt" is "2DS" for a 2D Static Model, "3DS" for a 3D Static Model, and "3DD" for a 3D Dynamic Model; and xxxxx is the model sequence number (not the SSDB model number). The Vertex Table File format shall be as follows:

for each vertex
  Vertex Record

5.1.2.2.3.3.1 Vertex Record. The number of these records shall correspond to the value in the Number of Model Vertices field within the Model Header record. The field structure of this record shall be as follows:

Coordinate Field

5.1.2.2.3.4 Data Source Table File. There shall be exactly one of these files in the SIF Model Section. The name of this file shall be "MODEL.DST". The Data Source Table File format shall be as follows:

SIF File Identifier Record
Data Source Table Header Record
for each data source table entry
  Data Source Table Entry Record

5.1.2.2.3.4.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI MODELS')
File Identifier Field (always 'DATA SOURCE TABLE')

5.1.2.2.3.4.2 Data Source Table Header Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'DS')
Number of Data Sources Field

5.1.2.2.3.4.3 Data Source Table Entry Record. The number of these records shall correspond to the count given in the header record. The field structure of this record shall be as follows:

Record Keyword Field (always 'SE')
Source ID Number Field
Source Type Field
Source Name Field
Source Date Field
Source Agency/Project Field
Reliability of Data Field
Accuracy Field
Collection System Field
Compilation Date Field
Compilation Criteria Field
Security Classification Field
Codewords Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.1.2.2.3.5 FACS Table File. The name of this file shall be "MODEL.FAC". The FACS Table File format shall be as follows:

SIF File Identifier Record
FACS Table Header Record
for each FACS table entry
  FACS Table Entry Record

5.1.2.2.3.5.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI MODELS')
File Identifier Field (always 'FACS TABLE')

5.1.2.2.3.5.2 FACS Table Header Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'FT')
Number of FACS Table Entries Field

5.1.2.2.3.5.3 FACS Table Entry Record. The number of these records shall correspond to the count given in the header record. The field structure of this record shall be as follows:

Record Keyword Field (always 'FE')
FACS Table Index Field
Number of FACS Attributes for This Entry Field
for each FACS attribute
  FACS Attribute Subrecord

5.1.2.2.3.5.3.1 FACS Attribute Subrecord. The field structure of each record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Attribute Value Field

5.1.2.2.3.5.3.2 FACS Application. When used to represent model attributes, FACS fields shall be applied at the level specified herein. Levels of applicability are abbreviated as follows: L = LOD; S = Point Light String; C = Component; and P = Polygon. Other levels are explicitly stated.

FACS Field Name	Application			
Absorptivity Field		S	C	P
Base Polygon ID	L			
Centroid Field (Deleted)	L	S		
Color Table Index Field				P
Cycle Rate Off Time Field		S	C	P
Cycle Rate On Time Field		S	C	P
Diffuse Reflectance Field		S	C	P
Directionality Field			C	P
Directivity (Infrared) Field		S		
Directivity (Radar) Field		S		
Directivity Field	L	S		
Emissivity Field		S	C	P
Exitance Field		S	C	P
Feature Identification (FID) Code Field	Model			
Feature Onset Field		S	C	P
Fixed Order Priority Field				P
Internal Material Category Field		S		
Internal Material Volume Field		S		
Layer Number (Infrared) Field		S	C	P
Layer Number (Radar) Field			C	P
Layer Number (Visual) Field			C	P
Light Horizontal Center Field		S	C	P
Light Horizontal Fall Field		S	C	P
Light Horizontal Width Field		S	C	P
Light Intensity Field		S	C	P
Light Type Field			C	P
Light Vertical Center Field		S	C	P
Light Vertical Fall Field		S	C	P
Light Vertical Width Field		S	C	P
Long Lineal Field		S		
Low Level Effects Field		S		
Maximum Edges Per Polygon Field	Cluster			
Maximum Height Field	L			
Model Centroid Field	L	S		
Object Volume Field		S		
Placement Point Field	L			
Polygon Illumination Type Field			C	P
Polygon Landing Light Illumination Field			C	P
Polygon Non-Occulting Field			C	P
Polygon Non-Shadow Field			C	P
Radius Field	L	S		
Reflectance		S	C	P
Self-Emitter Field		S	C	P
Shading Type Field			C	P
Shape Code Field		S		
Specular Field			C	P
Surface Material Category Field			C	P
Surface Material Subtype Field		S	C	P
Texture Map Reflectance Field		S		
Translucency Field			C	P
Transmissivity Field		S	C	P
Visible Range Field		S		

5.1.2.2.3.6 User-Defined FACS Table File. This file shall be included whenever the SIF data base contains nonstandard FACS codes. The name of this file shall be "MODEL.UFA". The format of this file shall be as follows:

```
SIF File Identifier Record
User-Defined FACS Table Header Record
for each user-defined FACS table entry
   User-Defined FACS Table Entry Record
```

5.1.2.2.3.6.1 SIF File Identifier Record. The field structure of this record shall be as follows:

```
Section Identifier Field (always 'SIF/HDI MODELS')
File Identifier Field (always 'USER-DEFINED FACS TABLE')
```

5.1.2.2.3.6.2 User-Defined FACS Table Header Record. The table shall be structured as follows:

```
Record Keyword Field (always 'UF')
Number of User-Defined FACS Attribute Codes Field
```

5.1.2.2.3.6.3 User-Defined FACS Table Entry Record. The table shall be structured as follows:

```
Record Keyword Field (always 'UE')
FACS Attribute Code Field
FACS Description Field
FACS Class Field
if FACS Class = ENUMERATED then
   Number of Enumerated Items Field
   for each Enumerated Item
      Enumerated Item Name Field
else
   Data Range Field
end if
```

5.1.2.2.3.7 Color Table File. The name of this file shall be "MODEL.CLR". The Color Table File format shall be as follows:

```
SIF File Identifier Record
Color Table Header Record
for each color table entry
      Color Table Entry Record
```

5.1.2.2.3.7.1 SIF File Identifier Record. The field structure of this record shall be as follows:

```
Section Identifier Field (always 'SIF/HDI MODELS')
File Identifier Field (always 'COLOR TABLE')
```

5.1.2.2.3.7.2 Color Table Header Record. The field structure of this record shall be as follows:

```
Record Keyword Field (always 'CT')
Color Definition Type Field
Number of Colors Field
```

5.1.2.2.3.7.3 Color Table Entry Record. The number of these records shall correspond to the count given in the header record. The field structure of this record shall be as follows:

Record Keyword Field (always 'CE')
Color Table Index Field
Color Description Field
RGB/HCV Color Value Field

5.1.2.2.3.8 Face-Based Texture Reference Table File. The name of this file shall be "MODEL.FTR". The Face-Based Texture Reference Table File format shall be as follows:

SIF File Identifier Record
Face-Based Texture Reference Table Header Record
for each texture reference
    Face-Based Texture Reference Record

5.1.2.2.3.8.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI MODELS')
File Identifier Field (always 'FACE-BASED TEXTURE REFERENCE TABLE')

5.1.2.2.3.8.2 Face-Based Texture Reference Table Header Record. The field structure of the record shall be as follows:

Record Keyword Field (always 'FX')
Number of Texture References Field

5.1.2.2.3.8.3 Face-Based Texture Reference Record. The field structure of the record shall be as follows:

Record Keyword Field (always 'FB')
Texture Reference Table Index Field
Texture Library Field
Texture ID Field
Texture Origin Field
Boundary ID Field
Mirror Field
Wrap Field
Wrap Type Field
Texture Scale Field
Polygon Alignment Vector Field
Rotation About Texture Origin Field
Polygon Reference Point Field
Layer Number Field

5.1.2.2.3.9 Vertex-to-Vertex Texture Reference Table File. There shall be one texture pattern vertex defined for each polygon vertex. The first texture pattern vertex shall map to the first polygon vertex. The name of this file shall be "MODEL.VTR" The Vertex-to-Vertex Texture Reference Table File format shall be as follows:

SIF File Identifier Record
Vertex-to-Vertex Texture Reference Table Header Record
for each texture reference
  Vertex-to-Vertex Texture Reference Record

5.1.2.2.3.9.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI MODELS')
File Identifier Field (always 'VERTEX-TO-VERTEX TEXTURE REFERENCE TABLE')

5.1.2.2.3.9.2 Vertex-to-Vertex Texture Reference Table Header Record. The field structure of the record shall be as follows:

Record Keyword Field (always 'VX')
Number of Texture References Field

5.1.2.2.3.9.3 Vertex-to-Vertex Texture Reference Record. There shall be one texture pattern vertex defined for each polygon vertex. The first texture pattern vertex shall map to the first polygon vertex. The field structure of the record shall be as follows:

Record Keyword Field (always 'VB')
Texture Reference Table Index Field
Texture Library Field
Texture ID Field
Layer Number Field
Number of Texture Pattern Coordinates Field
Texture Pattern Coordinates Subrecord

5.1.2.2.3.9.3.1 Texture Pattern Coordinates Subrecord. The field structure of the subrecord shall be as follows:

for each texture pattern point
  Texture Pattern Coordinates (X,Y) Field

5.1.2.2.3.10 Model-Based Texture Reference Table File. The name of this file shall be "MODEL.MTR". The Model-Based Texture Reference Table File format shall be as follows:

SIF File Identifier Record
Model-Based Texture Reference Table Header Record
for each texture reference
  Model-Based Texture Reference Record

5.1.2.2.3.10.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI MODELS')
File Identifier Field (always 'MODEL-BASED TEXTURE REFERENCE TABLE')

5.1.2.2.3.10.2 Model-Based Texture Reference Record. The field structure of the record shall be as follows:

Record Keyword Field (always 'MX')
Number of Texture References Field

5.1.2.2.3.10.3 Model-Based Texture Reference Record. The field structure of the record shall be as follows:

Record Keyword Field (always 'MB')
Texture Reference Table Index Field
Texture Library Field
Texture ID Field
Texture Origin Field
Boundary ID Field
Mirror Field
Wrap Field
Wrap Type Field
Texture Scale Field
Orientation Vectors Field
Model Reference Point Field
Layer Number Field

5.1.2.2.3.11 Non-Mapped Texture Reference Table File. The name of this file shall be "MODEL.NTR". The Non-Mapped Texture Reference Table File format shall be as follows:

SIF File Identifier Record
Non-Mapped Texture Reference Table Header Record
for each texture reference
    Non-Mapped Texture Reference Record

5.1.2.2.3.11.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI MODELS')
File Identifier Field (always 'NON-MAPPED TEXTURE REFERENCE TABLE')

5.1.2.2.3.11.2 Non-Mapped Texture Reference Table Header Record. The field structure of the record shall be as follows:

Record Keyword Field (always 'NX')
Number of Texture References Field

5.1.2.2.3.11.3 Non-Mapped Texture Reference Record. The number of these records shall correspond to the count given in the Non-Mapped Texture Reference Table Header Record. The field structure of the record shall be as follows:

Record Keyword Field (always 'NM')
Texture Reference Table Index Field
Texture Library Field
Texture ID Field

5.1.2.3 Culture Data. Producers of SIF/HDI culture shall transfer databases using either of two approaches to multiple levels of detail: layered or merged. The layered multi-LOD approach shall be used to represent multiple co-located culture tiles at different LODs. When layered culture data tiles are created, pointers (LOD Cross References) between related features in the lower resolution tiles and the higher resolution tiles shall be provided. The merged single-layer approach shall be used to represent a single layer of tiles throughout the gaming area. Each embedded patch of higher (or lower) resolution data shall be outlined and identified using island descriptor fields within the Data Resolution Identifier Record. Initially, the SDBF shall be responsible for segregating merged culture data into the SSDB layered LOD structure, and extracting requested SSDB data at the highest resolution available within the selected area of coverage and merging it into a single-layer culture database.

5.1.2.3.1 Culture Data Encoding. Comment fields or free text fields shall be embedded into a SIF ASCII data file as follows. The record keyword for a comment record shall be two consecutive asterisks (**). Following the keyword shall be the standard ASCII null character ('00') as a field separator. The comment field shall then continue until end of file (EOF) or the end of field separator is located in the SIF data file. Comment records shall not occur in the middle of any record in the file, but can be placed before or after any other record in the data file. Culture vertex data shall be encoded as binary values, but the headers and feature descriptors shall be encoded using a compressed form of ASCII. This compression shall take the form of stripping all leading zeros from numeric strings and all trailing blanks from character strings. Every ASCII field shall be separated from its neighbors by the ASCII null character ('00'). A SIF/HDI culture data set shall be comprised of six classes of features, defined as follows: Areal features are line segments which form a closed polygon around the area being described; linear (or lineal) features are line segments which typically do not form a closed polygon; point features consist of one or more discrete (non-connected) vertices; point light features consist of a single point which represents a light-emitting feature; point light string features are line segments consisting of two or more discrete (non-connected) vertices representing a light-emitting feature; and model references are a point location at which a model from the SIF/HDI model libraries may be inserted as a substitute for one or more culture features. For each of these classes of features there are certain rules which shall be followed, as defined below.

5.1.2.3.1.1 Areal Feature Rules. Given a SIF/HDI culture tile with the areal features shown in Figure 5, the following rules shall apply.

5.1.2.3.1.1.1 Background Feature. Areal feature 1 (F1) shall be the background feature, whose outline corresponds to the boundary of the tile.

5.1.2.3.1.1.2 Rendering Priority. Rendering priorities shall be specified via the layer number attribute associated with each feature, not sequence number.

5.1.2.3.1.1.3 Line Segments. Each areal feature shall consist of one or more line segments. Each segment shall consist of two or more vertex coordinates. A segment shall terminate whenever it is intersected by another segment (e.g., at V5 and V7).

5.1.2.3.1.1.4 Shared Segments. When a segment is shared by two or more features, it shall be stored only once in the database.

5.1.2.3.1.1.5 Feature Traversal. The vertices making up an areal feature shall be traversed in a counterclockwise direction as viewed from above. However, it is possible to list the vertices within a segment in a clockwise sequence. In the case of a segment shared by adjoining features, the sequence of vertices shall be clockwise relative to one of the features. For example, if the vertices in segment S5 are listed in the sequence V5, V6, V7, then this is counterclockwise relative to feature F2 but clockwise relative to feature F3. In order to support counterclockwise traversal of features in such situations, the Segment Pointer Record shall contain a traversal direction flag indicating that the vertices should be traversed in reverse sequence.

5.1.2.3.1.1.6 Closure. Areal features shall be explicitly closed.

5.1.2.3.1.1.7 Concave Features. There shall be no restriction against non-convex features in SIF/HDI. There shall be no restriction against use of SIF/HDI for databases in which concave features have been decomposed into convex polygons, but this use should be discouraged.

5.1.2.3.1.1.8 Inside Segments. It shall be possible to encode an areal feature with a "hole" within it by use of "inside" segments. For example, if F4 were not a feature in its own right but merely a hole within feature F3, then segment S8 shall be associated with feature F3 as an interior segment. A flag within the Segment Pointer Record shall be used to identify the segment as such.

5.1.2.3.1.1.9 Disjoint Polygons. It shall be possible to encode two or more disjoint polygons as a single areal feature. For example, features F5 and F6 could both be small ponds. To avoid redundant storage of feature attributes, it shall be possible to store one of the areal segments (say, S9) as the primary segment for feature F5, and store the remaining segment (S10) as a disjoint segment also within feature F5. A flag within the Segment Pointer Record shall be used to indicate such usage.

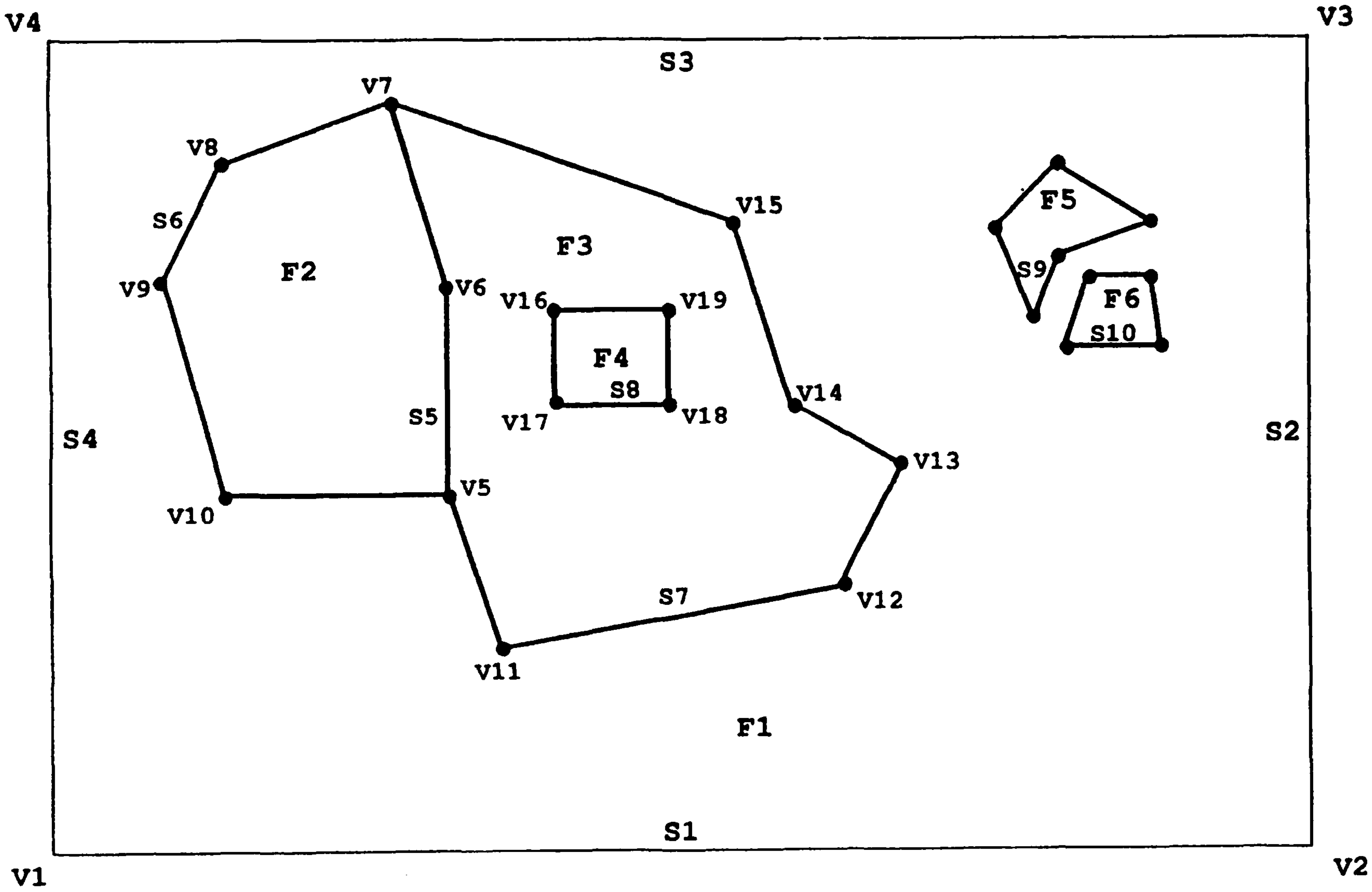


Figure 5. Areal Feature Conventions.

5.1.2.3.1.1.10 Non-redundant Vertices. Vertex coordinates shall be stored non-redundantly within one of two vertex files associated with a culture tile--a 2-D vertex file and a 3-D vertex file. For example, vertex V7 would be stored only once even though it is referenced by three segments (S5, S6, S7). Each segment header shall have a flag indicating whether the vertex coordinates may be found in the 2-D vertex file or the 3-D vertex file.

5.1.2.3.1.1.11 Vertex Ordering. Vertex coordinate records shall be referenced by their relative list position within a vertex file.

5.1.2.3.1.1.12 Feature/Segment Numbering. Feature and segment numbers shall be sequentially assigned, and explicitly encoded within feature and segment records. Each segment shall have a backpointer to the feature(s) which reference it, so that a two-way relationship can be maintained.

5.1.2.3.1.2 Linear Feature Rules. Given a SIF/HDI culture tile with the linear (also referred to as "lineal") features shown in Figure 6, the following rules shall apply.

5.1.2.3.1.2.1 Rendering Priority. Rendering priorities shall be specified via the layer number attribute associated with each feature, not the sequence number.

5.1.2.3.1.2.2 Line Segments. Each linear feature shall consist of one or more line segments. Each segment shall consist of two or more vertex coordinates. A segment shall be split into two segments whenever it is intersected by another segment. For example, vertex V3, which is the termination of feature F2 (V3 to V9) where it intersects with F1 (V1 to V5), is used to break up F1 into segments S1 (V1 to V3) and S2 (V3 to V5).

5.1.2.3.1.2.3 Segment Ends. Except for feature intersections, the definition of segment ends may be arbitrary. For example, feature F1 is shown as consisting of two segments, with S1 consisting of vertices V1, V2, and V3, and S2 consisting of vertices V3, V4, and V5; it would be perfectly acceptable to break either S1 or S2 (or both) into two segments containing two vertices each.

5.1.2.3.1.2.4 Shared Segments. When a segment is shared between a linear and an areal feature, it shall be stored only once in the database. For example, segment S4 (defined by vertices V7 and V8) is a common segment shared by linear feature F2 and areal feature F3.

5.1.2.3.1.2.5 Directionality. Uni-directional linears shall be digitized from left to right facing the visible/reflective side; i.e., the visible/reflective side shall be to the right as one traverses the vertex coordinates. For example, if F1 were a uni-directional feature with vertices listed in the sequence from V1 to V5, then the visible/reflective side would be towards the bottom of the diagram.

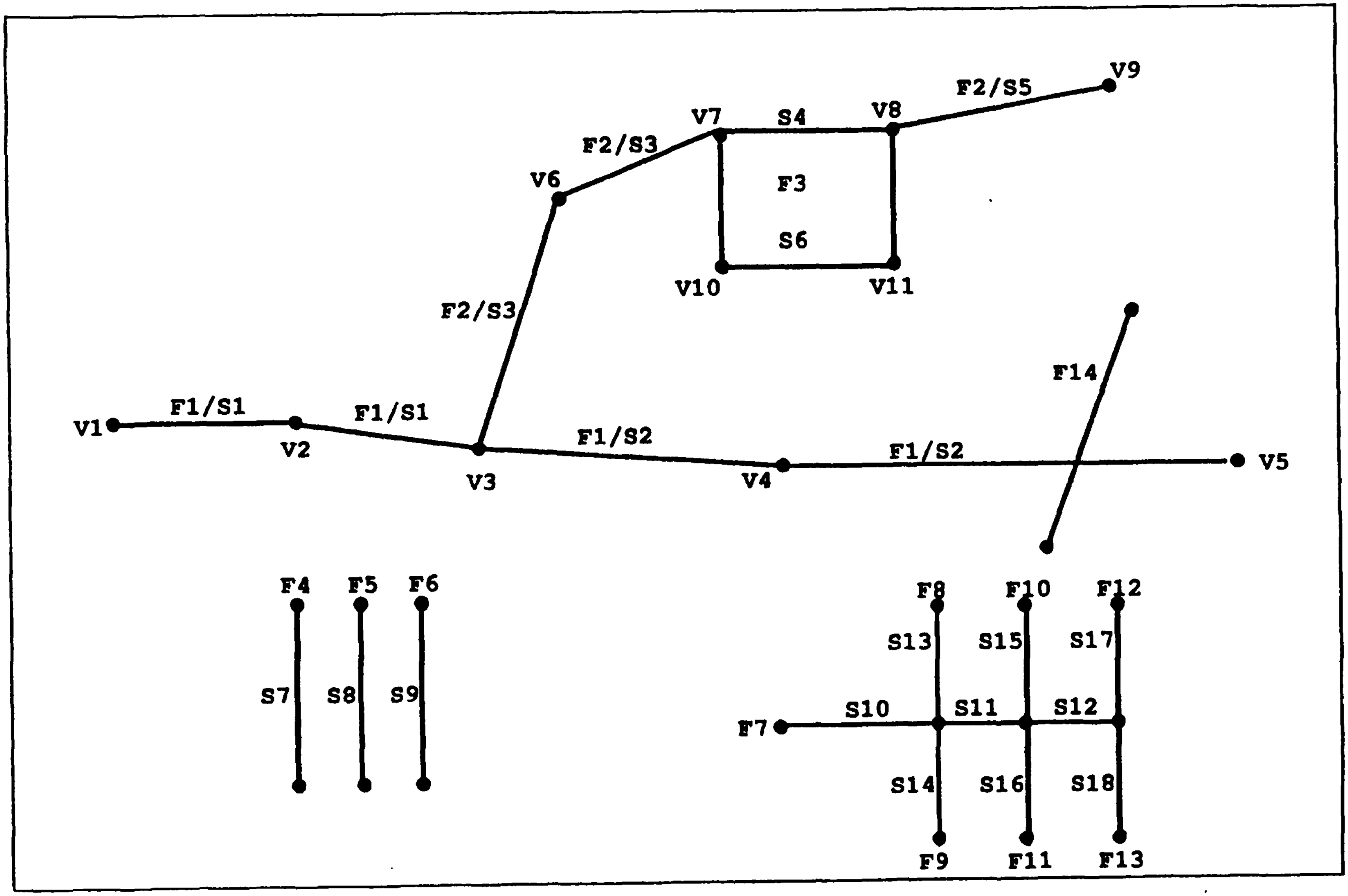


Figure 6. Linear Feature Conventions.

5.1.2.3.1.2.6 Feature Traversal. Except for uni-directional features, either end of a linear feature or segment may serve as the beginning point for traversal. Once the direction is decided for a specific feature, the segments making up a linear feature shall be traversed continuously in that direction. It is possible to list the vertices within a segment in a sequence opposite to the direction of traversal within the overall linear feature. This situation is most likely to arise in the case of a segment shared with an adjoining areal feature. For example, the linear feature F2 happens to share segment S4 with areal feature F3. If F3 happened to have been digitized prior to F2, segment S4 would normally be digitized in the direction V8 to V7, in order to maintain the rule of counter-clockwise traversal of areals. From the standpoint of linear feature F2, however, this vertex sequence is opposite to the primary flow from V3 to V9. In order to support continuous traversal of linear features in such situations, the Segment Pointer Record shall contain a traversal direction flag indicating that the vertices shall be traversed in reverse sequence.

5.1.2.3.1.2.7 Disjoint Segments. It shall be possible to encode two or more disjoint line segments as a single linear feature. For example, features F4, F5, and F6 could be rows of hedges. To avoid redundant storage of feature attributes, it would be possible to store one of the linear segments (such as S7) as the primary segment for feature F4, and store the remaining segments (S8 and S9) as disjoint segments also within feature F4. A flag in the Segment Pointer Record shall be used to indicate such usage.

5.1.2.3.1.2.8 Non-contiguous Feature. It shall be possible to encode two or more connected but non-continuous line segments as a single linear feature. For a more compressed representation, it shall be possible to store one set of continuous line segments (say, S10, S11, S12) as the primary component of feature F7, and store the remaining segments (S13 through S18) as disjoint segments also within feature F7. A flag in the Segment Pointer Record shall be used to indicate such usage.

5.1.2.3.1.2.9 Non-redundant Vertices. Vertex coordinates shall be stored non-redundantly within one of two vertex files associated with a culture tile--a 2-D vertex file and a 3-D vertex file. For example, vertex V3 would be stored only once even though it is referenced by three segments (S1, S2, S3). Each segment header shall have a flag indicating whether the vertex coordinates may be found in the 2-D vertex file or the 3-D vertex file.

5.1.2.3.1.2.10 Vertex Ordering. Vertex coordinate records shall be referenced by their relative list position within a vertex file.

5.1.2.3.1.2.11 Feature/Segment Numbering. Feature and segment numbers shall be sequentially assigned, and explicitly encoded within feature and segment records. Each segment shall have a backpointer to the feature(s) which reference it, so that a two-way relationship can be maintained.

5.1.2.3.1.3 Point Feature Rules. Given a SIF/HDI culture tile with the point features shown in Figure 7, the following rules shall apply.

5.1.2.3.1.3.1 Rendering Priority. Rendering priorities shall be specified via the layer number attribute associated with each feature, not the sequence number.

5.1.2.3.1.3.2 Line Segments. Each point feature shall be represented as a segment consisting of one or more vertex[4] coordinates. Multiple vertices shall be used to encode a series of discrete point objects having common attributes (e.g., power pylons). Feature F1 illustrates the more common single-vertex case. Features F2 and F3 illustrate two varieties of the multi-vertex case.

5.1.2.3.1.3.3 Vertex Sequence. As shown in F3, when multiple vertices are specified within a point feature, their sequence can be arbitrary; i.e., it is not required that they represent a direction of traversal.

5.1.2.3.1.3.4 Disjoint Segments. If multi-vertex point feature is encoded using multiple segments, segments other than the first shall be flagged as disjoint segments.

5.1.2.3.1.3.5 Coincident Segments. It shall be possible for a point feature to be located on an areal or lineal feature segment. In such cases, the point feature vertex shall serve as an end node defining the break point between two line segments. For example, if point feature F4 were to actually lie upon line segment S5 of linear feature F5, then segment S5 should be split into two segments at the point at which F4 intersects F5. V11 would become a vertex defining feature F5; feature F5 would then consist of two segments, one containing vertices V12, V13, and V11, and the other containing V11, V14, and V15. While vertex V11 becomes shared by F4 and F5, segment S4 remains applicable only to point feature F4 and shall not become a segment within lineal feature F5.

5.1.2.3.1.3.6 Non-redundant Vertices. Vertex coordinates shall be stored non-redundantly within one of two vertex files associated with a culture tile. For example, vertex V11 shall be stored only once, even if it were to be referenced by point feature F4 and by two line segments within lineal feature F5. Each segment header shall include a flag indicating whether the vertex coordinates may be found in the 2-D vertex file or the 3-D vertex file.

5.1.2.3.1.3.7 Vertex Ordering. Vertex coordinate records shall be referenced by their relative list position within a vertex file.

5.1.2.3.1.3.8 Feature Numbering. Feature numbers shall be sequentially assigned, and explicitly encoded within feature records.

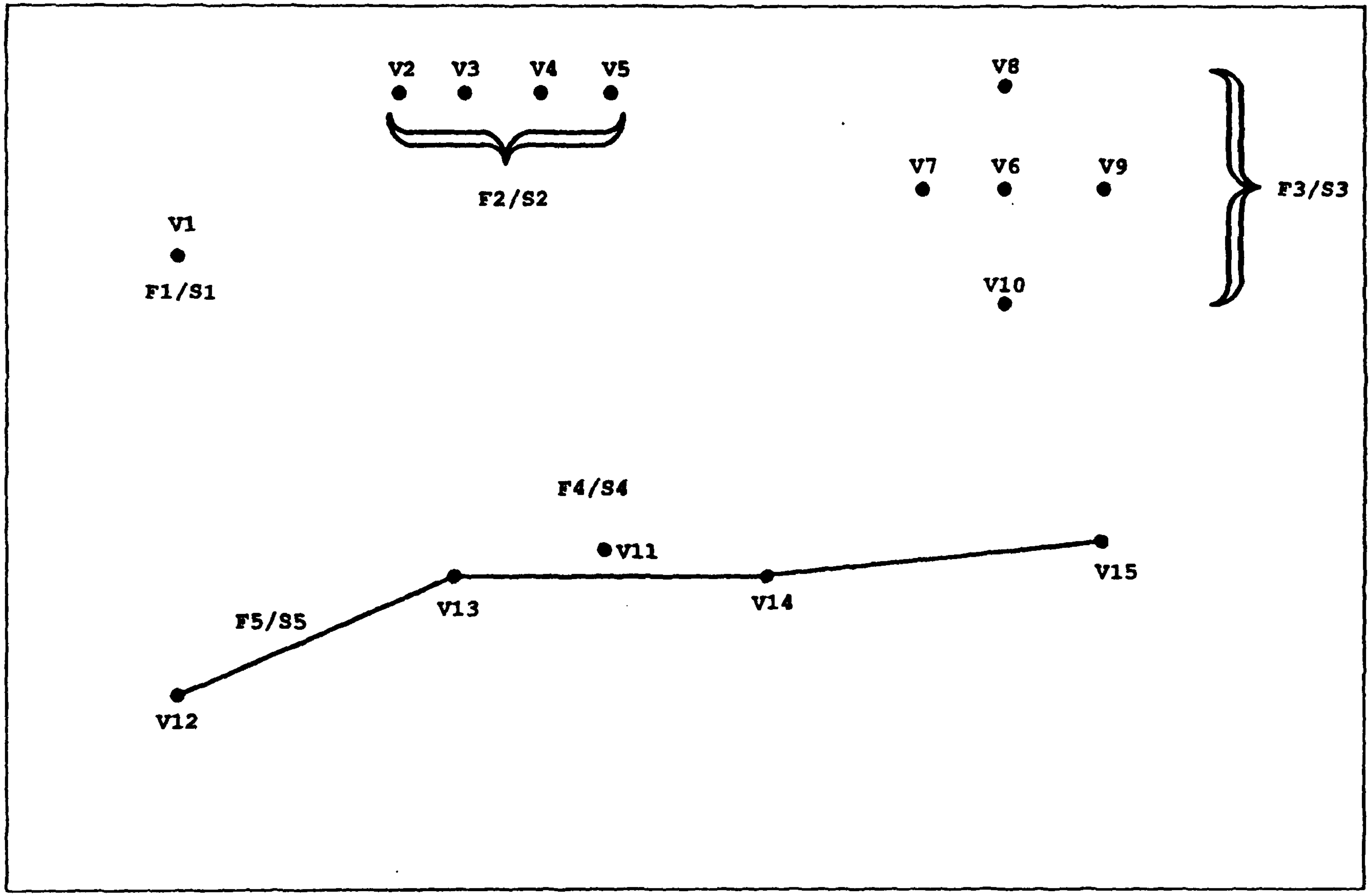


Figure 7. Point Feature Conventions.

5.1.2.3.1.4 Point Light Feature Rules. Given a SIF/HDI culture tile with the point light features shown in Figure 8, the following rules shall apply.

5.1.2.3.1.4.1 Rendering Priority. Rendering priorities shall be specified via the layer number attribute associated with each feature, not the sequence number.

5.1.2.3.1.4.2 Number of Vertices. As illustrated by feature F1, each point light feature shall be a segment consisting of one and only one vertex coordinate. (Features consisting of multiple point lights shall be encoded as point light string features.)

5.1.2.3.1.4.3 Coincident Segments. It shall be possible for a point light feature to be located on an areal or lineal feature segment. In such cases, the point light feature vertex shall serve as an end node defining the break point between two line segments. For example, if point light feature F2 were to lie upon line segment S3 of linear feature F3, then segment S3 should be split into two segments at the point at which F2 intersects F3. V2 would become a vertex defining feature F3; feature F3 would then consist of two segments, one containing vertices V3, V4, and V2, and the other containing V2, V5, and V6. While vertex V2 becomes shared by F2 and F3, segment S2 remains applicable only to point light feature F2 and shall not become a segment within lineal feature F3.

5.1.2.3.1.4.4 Non-redundant Vertices. Vertex coordinates shall be stored non-redundantly within one of two vertex files associated with a culture tile. For example, vertex V2 shall be stored only once even if it were to be referenced by point light feature F2 and by two line segments within lineal feature F3. Each segment header shall include a flag indicating whether the vertex coordinates may be found in the 2-D vertex file or the 3-D vertex file.

5.1.2.3.1.4.5 Vertex Ordering. Vertex coordinate records shall be referenced by their relative list position within a vertex file.

5.1.2.3.1.4.6 Feature Numbering. Feature numbers shall be sequentially assigned, and encoded within feature records.

5.1.2.3.1.5 Point Light String Feature Rules. Given a SIF/HDI culture tile with the point light string features shown in Figure 9, the following rules shall apply.

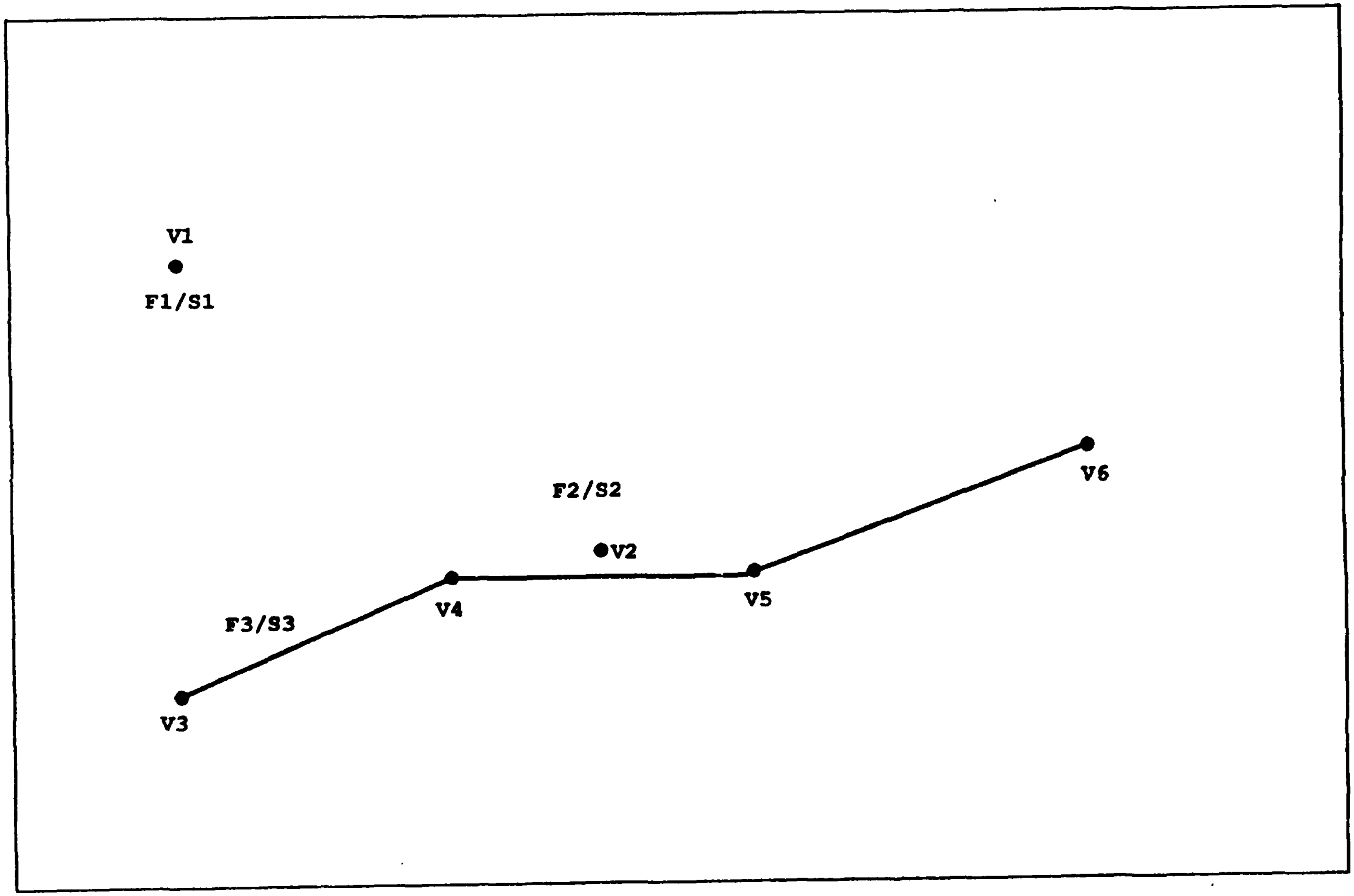


Figure 8. <u>Point Light Feature Conventions</u>.

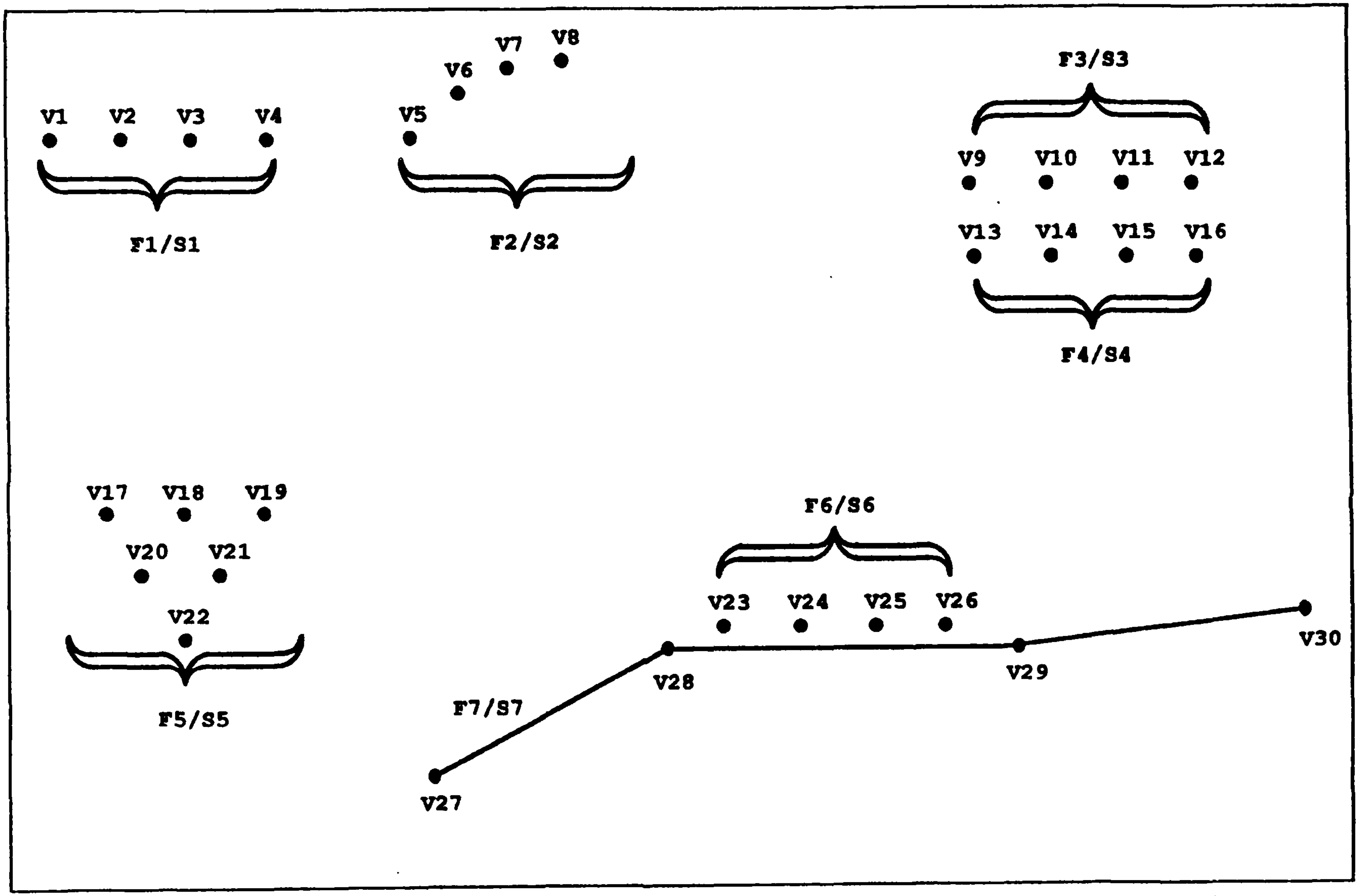


Figure 9. Point Light String Feature Conventions.

5.1.2.3.1.5.1 Rendering Priority. Rendering priorities shall be specified via the layer number attribute associated with each feature, not the sequence number.

5.1.2.3.1.5.2 Number of Vertices. Each point light string feature shall be a segment consisting of two or more vertices. Point light string records shall be used to encode a series of discrete point lights having common attributes (such as runway lights). Feature F1 illustrates the most common type of point light string, a series of lights arranged in a straight line. Vertex coordinates shall be specified for each light in the string.

5.1.2.3.1.5.3 Non-linear Strings. Feature F2 shows a series of lights arranged in a curved line. In such cases, each point light location shall be specified as an explicit vertex.

5.1.2.3.1.5.4 Parallel Strings. Features F3 and F4 represent two parallel straight-line rows of lights. Instead of encoding each row as a separate point light string feature, it is allowable to encode two or more strings having common attributes as a single feature, every light as a separate vertex within the feature segment, or by specifying one row of lights as the primary segment of the feature, and the second row as a disjoint segment of the same feature.

5.1.2.3.1.5.5 Light Groups. The point light string record shall be used to encode a feature consisting of a group of point lights having common attributes but arranged in non-linear fashion. Feature F5 is an example.

5.1.2.3.1.5.6 Vertex Sequence. In all cases where the point light vertices are explicitly listed, the sequence of vertex coordinates may be arbitrary.

5.1.2.3.1.5.7 Coincident Segments. It shall be possible for a point light string feature to be co-located with an areal or lineal feature segment. In such cases, the point light string feature vertices also serve as end nodes defining the intersection of two line segments. For example, if point light string feature F6 were to lie upon line segment S7 of linear feature F7, then segment S7 shall be split into five segments at the points at which F6 intersects F7. Vertices V23, V24, V25, and V26 would become vertices defining feature F7; feature F7 would then consist of five segments; the first would contain vertices V27, V28, and V23, the second would contain V23 and V24, the third V24 and V25, the fourth V25 and V26, and the fifth V26, V29, and V30. While vertices V23 through V26 becomes shared by F6 and F7, segment S6 remains applicable only to point light string feature F6 and shall not become a segment within lineal feature F7.

5.1.2.3.1.5.8 Non-redundant Vertices. Vertex coordinates shall be stored non-redundantly within one of two vertex files associated with a culture tile. For example, vertex V23 shall be stored only once even if it were to be referenced by point light string feature F6 and by two line segments within lineal feature F7. Each segment header shall include a flag indicating whether the vertex coordinates may be found in the 2-D vertex file or the 3-D vertex file.

5.1.2.3.1.5.9 Vertex Ordering. Vertex coordinate records shall be referenced by their relative list position within a vertex file.

5.1.2.3.1.5.10 Feature Numbering. Feature numbers shall be sequentially assigned, and explicitly encoded within feature records.

5.1.2.3.1.6 Model Reference Rules. Model references shall be used to permit discretionary substitution of culture features by models from a model library. The following rules shall apply to the use of model references.

5.1.2.3.1.6.1 Number of Vertices. Each model reference shall contain a model identifier along with one and only one vertex coordinate. This coordinate shall represent the reference point, relative to the southwest corner of the tile, used to place and orient a model when inserted into the database. Within the geometry of every model there shall be a Placement Point which is used to align the model to the model reference coordinate. Along with the reference coordinate, each model reference shall include an orientation angle and a scaling factor.

5.1.2.3.1.6.2 Model Reference Table. To indicate that a particular feature may be replaced by a particular model, two database entries shall be made. First, an entry shall be made in the Model Reference Table identifying the model and its placement instructions. Second, for the culture feature being replaced by that model, a Model Reference Pointer record shall be added pointing to the model reference table entry.

5.1.2.3.1.6.3 Multiple References. When a single model replaces multiple cultural features, there shall be one entry in the Model Reference table, and a model reference pointer shall be added to every feature being substituted.

5.1.2.3.1.6.4 Multiple Models. A culture feature may have more than one model reference pointer.

5.1.2.3.1.6.5 Placement Coordinate. The model reference placement coordinate may be either 2-D or 3-D.

5.1.2.3.1.6.6 Table ID. Model reference table entries shall be referenced by ID numbers which are sequentially assigned.

5.1.2.3.1.7 Superfeature Rules. The primary use for superfeatures shall be to aggregate individual features within a culture tile into larger homogeneous data groups. The superfeature shall identify all child features that belong to this homogeneous group, and may indicate special "aggregate" features for the group. The data structures for the superfeatures have been defined with the capabilities for future flexibility. This flexibility has driven the relationships that are identified in the following paragraphs.

5.1.2.3.1.7.1 Child Feature References. A superfeature may reference one or more child features. A child feature is considered to be one feature of many that define the superfeature's group. For example, there may be many contiguous 3-D tree canopy features within the culture tile. A superfeature could be created which points at all of the 3-D features. Since each referenced feature would then define only a portion of the complete superfeature, the feature would be considered to be a child feature. There are no restrictions on the types or dimensions of features that may be referenced. This means that all different feature types (Areal, Linear, Point, Point Light, and Point Light Strings) can be grouped together into a superfeature, and that two-dimensional features may also be grouped together with three-dimensional features to form a superfeature.

5.1.2.3.1.7.1.1 To prevent a limitation on the expandability of superfeatures, and to allow the superfeature categorization to be truly user-defined, there can be more than one superfeature associated with a feature.

5.1.2.3.1.7.2 "Aggregate" Feature References. An "aggregate" feature is a special feature that is referenced by the superfeature. This feature can be considered to be a replacement for all of the children features being referenced. For example, a superfeature could reference many contiguous 3-D tree canopy polygons as well as reference a 2-D tree canopy feature that defines the outline of all of the 3-D contiguous canopy polygons. In this case, the 2-D polygon can be considered an "aggregate" feature since it can be used to describe the spatial extent of the tree canopy. See Figure 10.

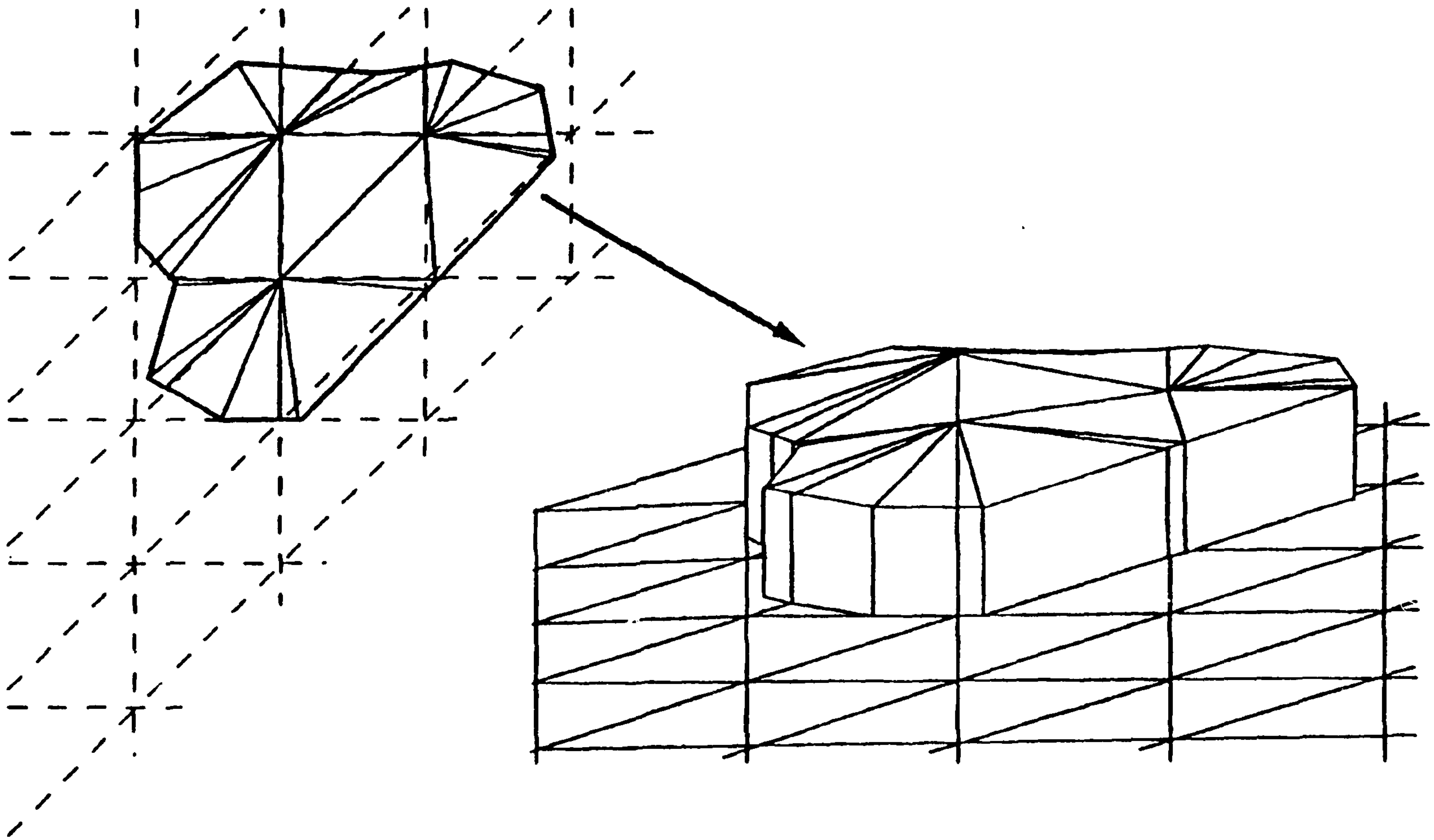
5.1.2.3.1.7.2.1 To prevent a limitation on the expandability of aggregate features, and to allow the superfeature categorization to be truly user-defined, there can be more than one aggregate feature associated with a superfeature.

5.1.2.3.1.7.3 Additional References. To provide for future expandability in the SIF standard, the following superfeature references shall be provided for in SIF data, however, the SSDB currently will not store these additional references.

5.1.2.3.1.7.3.1 A superfeature may reference one or more superfeatures. This means that a superfeature can be used as a subset (or child superfeature) of another superfeature. For example, it would be possible to take several superfeatures that identify individual but spatially related tree canopy features and combine them via a parent "forest" superfeature.

5.1.2.3.1.7.3.2 To prevent a limitation on the expandability of superfeatures, and to allow the superfeature categorization to be truly user-defined, there can be more than one parent superfeature associated with a superfeature.

5.1.2.3.1.7.3.3 To permit additional flexibility within the categorization scheme for the superfeatures, a combination of both features and superfeatures may be referenced by any given superfeature.



The feature on the left (the bold outline), is the 2-D representation of a tree canopy, or "solid" (polygonal) forested area. In this case it is also the "aggregate" feature referenced by a superfeature, which could replace the entire superfeature. The feature(s) on the right are the side and top 3-D canopy polygons which are created when the 2-D "aggregate" feature is fragmented on the terrain polygons (shown dashed on the left and solid on the right "under" the canopy polygon(s).

Figure 10. Example of aggregate superfeature.

5.1.2.3.2 Culture Section Structure. The SIF culture section shall be organized into cells of coverage delimited by full degree boundaries. Each cell of data shall contain multiple manuscripts at up to six levels of detail (LODs). The physical tape format shall have a fixed record size. Data fields and logical records may vary in length and be packed into the physical records. All records (except the file identifier record and table entry records) will begin with a 2-character keyword identifier. The SIF/HDI culture section format shall be as follows, and as shown in Figure 11.

For each culture database
  Database Header File
  Tile Information File
  For each culture tile
    Two-D Coordinate File [optional]
    Three-D Coordinate File [optional]
    FACS Table File [optional]
    User-Defined FACS Table File [optional]
    Color Table File [optional]
    FID/FDC Cross-Reference Table File [optional]
    Global-Based Texture Reference Table File [optional]
    Non-Mapped Texture Reference Table File [optional]
    Model Reference Table File [optional]
    Superfeature File [optional]
    Feature File
    Segment File

5.1.2.3.2.1 Database Header File. The name of this file shall be "CULTURE.DBH". The Database Header File format shall be as follows and as shown in Figure 12.

SIF File Identifier Record
SIF/HDI Culture Database Header Record
for each Data Source Table entry
  Data Source Table Record
  for each Accuracy Sub-region
    Accuracy Region Record [optional]

5.1.2.3.2.1.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI CULTURE')
File Identifier Field (always 'CULTURE DATABASE HEADER')

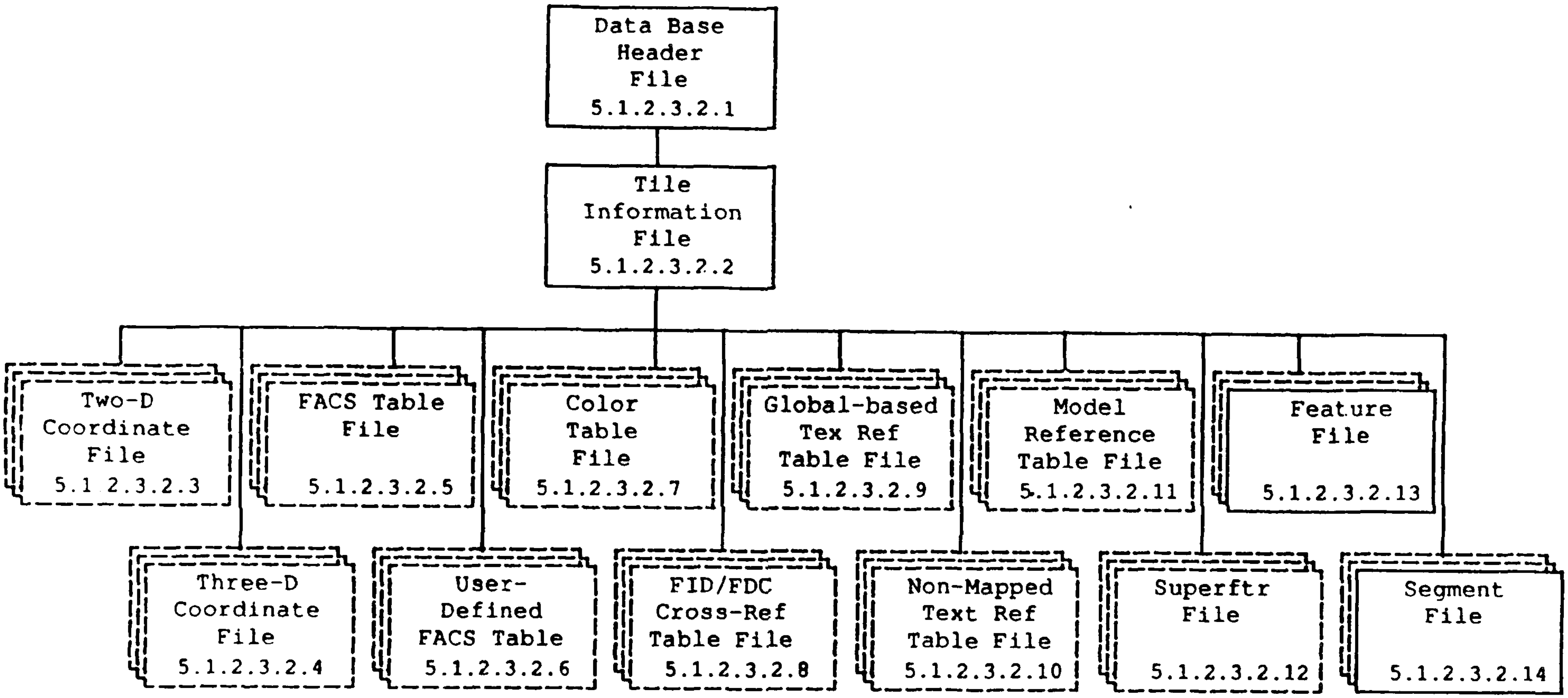


Figure 11. SIF/HDI Culture Data File Relationships.

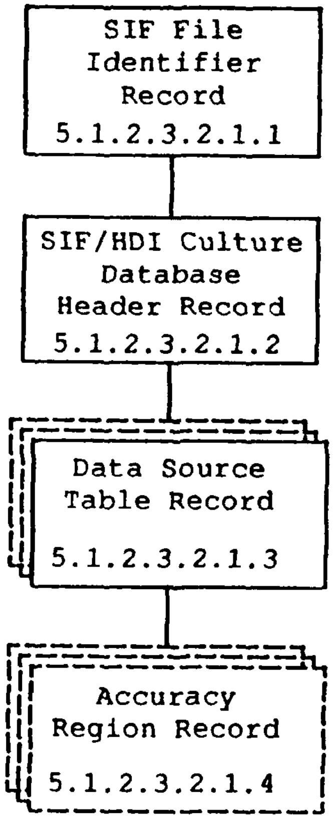


Figure 12. SIF/HDI Culture Database Header File's Record Relationships.

5.1.2.3.2.1.2 SIF/HDI Culture Database Header Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'DH')
Security Level Field
Culture Coordinate System Field ('GEODETIC' by convention)
Counter-Clockwise Areals Flag Field ('TRUE' by convention)
Explicit Closure of Areals Flag Field ('TRUE' by convention)
Number of LODs Field
Number of Tiles Field
Number of Database Boundary Coordinates Field
for each boundary coordinate
    Latitude/Longitude Field
Number Of Data Sources Field

5.1.2.3.2.1.3 Data Source Table Record. The number of these records associated with the transmitted manuscripts shall correspond to the value in the Number of Data Sources Field in the parent SIF/HDI Culture Database Header Record. The field structure of the Data Source Table Record shall be as follows:

Record Keyword Field (always 'DS')
Number of Accuracy Regions Field
Source ID Number Field
Source Type Field
Source Name Field
Source Date Field
Source Agency/Project Field
Data Edition Number Field
Data Series Designator Field
Producer Code Field
Reliability of Data Field
Relative Vertical Accuracy Field
Absolute Vertical Accuracy Field
Relative Horizontal Accuracy Field
Absolute Horizontal Accuracy Field
Collection System Field
Compilation Date Field
Compilation Criteria Field
Security Classification Field
Codewords Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.1.2.3.2.1.4 Accuracy Region Record. The number of accuracy regions defined shall correspond to the value in the Number of Accuracy Regions Field in the parent Data Source Table Record. The field structure of this subrecord shall be as follows:

Record Keyword Field (always 'AR')
Number of Boundary Points Field
Relative Vertical Accuracy Field
Absolute Vertical Accuracy Field
Relative Horizontal Accuracy Field
Absolute Horizontal Accuracy Field
for each boundary point
   Relative Coordinate Field

5.1.2.3.2.2 Tile Information File. The name of this file shall be "CULTURE.THI". The Tile Information File format shall be as follows:

SIF File Identifier Record
for each culture Tile
   Tile Header Record
   Data Resolution Record

5.1.2.3.2.2.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI CULTURE')
File Identifier Field (always 'CULTURE TILE INFORMATION')

5.1.2.3.2.2.2 Tile Header Record. There shall be one Tile Header Record per tile (manuscript) contained in the database. The coordinates that define the tile boundary should be stored in a "counter-clockwise" direction with explicit closure of the boundary polygon. The boundary polygon for each tile shall be defined as the minimum bounding geodetic rectangle that encompasses all of the data for the tile. The field structure of this record shall be as follows:

Record Keyword Field (always 'TH')
Manuscript ID Field
Number of Manuscript Boundary Coordinates Field
for each boundary coordinate
      Latitude/Longitude Field

5.1.2.3.2.2.3 Data Resolution Identifier Record. There shall be one Data Resolution Identifier record per culture tile contained in the database. The coordinates that define the boundary of a higher resolution island shall be stored in a "counter-clockwise" direction with explicit closure of the boundary polygon. The field structure of this record shall be as follows:

Record Keyword Field (always 'DR')
SSDB LOD Number Field
Default Source Identifier Field
Synthetic Data Flag Field
Number of Embedded Higher-Resolution Islands Field
for each island
    Island Number Field
    SSDB LOD Number Field
    Default Source Identifier Field
    Synthetic Data Flag Field
    Number of Island Boundary Coordinates Field
    for each boundary coordinate
        Latitude/Longitude Field

5.1.2.3.2.3 Two-D Coordinate File. There shall be one of these files per tile in the SIF database. The overall format of the Two-D Coordinate File shall be a binary file, where the coordinates shall be expressed as 32-bit signed integers. The name of this file shall be "CULrxxxxx.2DC", where "r" is "M" for Merged Culture Data, "0" for LOD 0 Culture Data, "1" for LOD 1 Culture Data, "2" for LOD 2 Culture Data, "3" for LOD 3 Culture Data, "4" for LOD 4 Culture Data, and "5" for LOD 5 Culture Data; and "xxxxx" is the culture tile sequence number. The Two-D Coordinate File format shall be as follows:

for each coordinate pair
    2-D Coordinate Record

5.1.2.3.2.3.1 2-D Coordinate Record. Each record in this file shall contain a 2-D geographic coordinate. Each coordinate shall contain a latitude and a longitude, expressed in resolution units of ten thousandths of arc seconds relative to the southwest corner of the tile. The field structure of this record shall be as follows:

Relative Latitude Field
Relative Longitude Field

5.1.2.3.2.4 Three-D Coordinate File. There shall be one of these files per tile in the SIF database. The overall format of the Three-D Coordinate File shall be a binary file, where the coordinates shall be expressed as 32-bit signed integers. The name of this file shall be "CULrxxxxx.3DC", where "r" is "M" for Merged Culture Data, "0" for LOD 0 Culture Data, "1" for LOD 1 Culture Data, "2" for LOD 2 Culture Data, "3" for LOD 3 Culture Data, "4" for LOD 4 Culture Data, and "5" for LOD 5 Culture Data; and "xxxxx" is the culture tile sequence number. The Three-D Coordinate File format shall be as follows:

for each coordinate triplet
    3-D Coordinate Record

5.1.2.3.2.4.1 3-D Coordinate Record. Each record in this file shall contain a 3-D geographic coordinate. Each coordinate shall contain a latitude and a longitude, expressed in resolution units of ten thousandths of arc seconds relative to the southwest corner of the tile, and an elevation, expressed in resolution units of 0.001 meters relative to Mean Sea Level. The field structure of this record shall be as follows:

Relative Latitude Field
Relative Longitude Field
Elevation Field

5.1.2.3.2.5 FACS Table File. There shall be zero or one of these files associated with each culture tile. A FACS table will be included with any tile that requires any of the optional descriptors associated with a feature. The name of this file shall be "CULrxxxxx.FAC", where "r" is "M" for Merged Culture Data, "0" for LOD 0 Culture Data, "1" for LOD 1 Culture Data, "2" for LOD 2 Culture Data, "3" for LOD 3 Culture Data, "4" for LOD 4 Culture Data, and "5" for LOD 5 Culture Data; and "xxxxx" is the culture tile sequence number. The FACS table file format shall be as follows:

SIF File Identifier Record
FACS Table Header Record
for each FACS Table Entry
  FACS Table Entry Record

5.1.2.3.2.5.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always '*SIF/HDI CULTURE*')
File Identifier Field (always 'FACS TABLE')

5.1.2.3.2.5.2 FACS Table Header Record. The FACS Table Header shall be structured as follows:

Record Keyword Field (always 'FT')
Number of FACS Table Entries Field

5.1.2.3.2.5.3 FACS Table Entry Record. The total number of these records shall correspond to the value in the file header record. Each FACS Table Entry shall be structured as follows:

Record Keyword Field (always 'FE')
FACS Table Index Field
Number of FACS Records for this Entry Field
for each FACS Record
  FACS Attribute Subrecord

5.1.2.3.2.5.3.1 FACS Attribute Subrecord. The field structure of this record shall be as follows:

FACS Class Field
FACS Attribute Code Field
Synthetic Data Flag Field
Source ID Number Field
Sensors Supported Field
Attribute Value Field

5.1.2.3.2.5.3.2 FACS Application. When used to represent feature attributes, FACS fields shall be used as specified herein. Applications are abbreviated as follows: A = Areal; L = Linear; P = Point; T = Point Light; S = Point Light String.

FACS Field Name	Application				
Absorptivity Field	A	L	P	T	S
Centroid Field (Deleted)	A				S
Culture Centroid Field	A				S
Cycle Rate Off Time Field				T	S
Cycle Rate On Time Field				T	S
Diffuse Reflectance Field	A	L	P	T	S
Directivity (Infrared) Field	A	L	P	T	S
Directivity (Radar) Field	A	L	P	T	S
Directivity Field	A	L	P	T	S
Emissivity Field	A	L	P	T	S
Exitance Field	A	L	P	T	S
Feature Onset Field	A	L	P	T	S
Internal Material Category Field	A	L	P	T	S
Internal Material Volume Field	A	L	P	T	S
Layer Number (Infrared) Field	A	L	P	T	S
Layer Number (Radar) Field	A	L	P	T	S
Light Horizontal Center Field				T	S
Light Horizontal Fall Field				T	S
Light Horizontal Width Field				T	S
Light Intensity Field				T	S
Light Vertical Center Field				T	S
Light Vertical Fall Field				T	S
Light Vertical Width Field				T	S
Long Lineal Field			P	T	S
Low Level Effects Field	A	L	P	T	S
Monitor Type Field	A				
Number of Structures Field	A				
Object Volume Field	A	L	P	T	S
Percent of Roof Coverage Field	A				
Percent of Tree Coverage Field	A				
Polygon Illumination Type Field	A				
Polygon Normal Field	A				
Radius Field	A	L	P	T	S
Reflectance Field	A	L	P	T	S
Roof Type Field	A				
Self-Emitter Field	A	L	P	T	S
Shading Type Field	A				
Shape Code Field	A		P		S
Specular Field	A	L	P		
Superfeature ID	A	L	P	T	S
Surface Material Subtype Field	A	L	P	T	S
Texture Map Reflectance Field	A	L	P	T	S
Translucency Field	A	L	P		
Transmissivity Field	A	L	P	T	S
Visible Range Field				T	S

5.1.2.3.2.6 User-Defined FACS Table File. The name of this file shall be "CULrxxxxx.UFA", where "r" is "M" for Merged Culture Data, "0" for LOD 0 Culture Data, "1" for LOD 1 Culture Data, "2" for LOD 2 Culture Data, "3" for LOD 3 Culture Data, "4" for LOD 4 Culture Data, and "5" for LOD 5 Culture Data; and "xxxxx" is the culture tile sequence number. The User-Defined FACS Table File format shall be as follows:

```
SIF File Identifier Record
User-Defined FACS Table Header Record
for each User-Defined FACS Table Entry
   User-Defined FACS Table Entry Record
```

5.1.2.3.2.6.1 SIF File Identifier Record. The field structure of this record shall be as follows:

```
Section Identifier Field (always 'SIF/HDI CULTURE')
File Identifier Field (always 'USER-DEFINED FACS TABLE')
```

5.1.2.3.2.6.2 User-Defined FACS Table Header Record. The User-Defined FACS Table Header shall be structured as follows:

```
Record Keyword Field (always 'UF')
Number of User-Defined FACS Attribute Codes Field
```

5.1.2.3.2.6.3 User-Defined FACS Table Entry Record. The total number of these records in the data file shall correspond to the value in the file header record. Each User-Defined FACS Table Entry shall be structured as follows:

```
Record Keyword Field (always 'UE')
FACS Attribute Code Field
FACS Description Field
FACS Class Field
if FACS Class = ENUMERATED then
   Number of Enumerated Items Field
   for each Enumerated Item
      Enumerated Item Name Field
else
   Data Range Field
end if
```

5.1.2.3.2.7 Color Table File. The name of this file shall be "CULrxxxxx.CLR", where "r" is "M" for Merged Culture Data, "0" for LOD 0 Culture Data, "1" for LOD 1 Culture Data, "2" for LOD 2 Culture Data, "3" for LOD 3 Culture Data, "4" for LOD 4 Culture Data, and "5" for LOD 5 Culture Data; and "xxxxx" is the culture tile sequence number. The Color Table File format shall be as follows:

```
SIF File Identifier Record
Color Table Header Record
for each Color Table Entry
   Color Table Entry Record
```

5.1.2.3.2.7.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI CULTURE')
File Identifier Field (always 'COLOR TABLE')

5.1.2.3.2.7.2 Color Table Header Record. The Color Table Header shall be structured as follows:

Record Keyword Field (always 'CT')
Color Definition Type Field
Number of Colors Field

5.1.2.3.2.7.3 Color Table Entry Record. The color table entry shall be structured as follows:

Record Keyword Field (always 'CE')
Color Table Index Field
Color Description Field
RGB/HCV Color Value Field

5.1.2.3.2.8 FID/FDC Cross-Reference Table File. This table shall be included if there are any user defined FID codes that do not map directly to SIF supported FDC codes. The name of this file shall be "CULrxxxxx.FFT", where "r" is "M" for Merged Culture Data, "0" for LOD 0 Culture Data, "1" for LOD 1 Culture Data, "2" for LOD 2 Culture Data, "3" for LOD 3 Culture Data, "4" for LOD 4 Culture Data, and "5" for LOD 5 Culture Data; and "xxxxx" is the culture tile sequence number. The FID/FDC Cross-Reference Table File format shall be as follows:

SIF File Identifier Record
FID/FDC Cross-Reference Header Record
for each FID/FDC Cross Reference Table Entry
  FID/FDC Cross-Reference Table Entry Record

5.1.2.3.2.8.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI CULTURE')
File Identifier Field (always 'FID/FDC CROSS REFERENCE TABLE')

5.1.2.3.2.8.2 FID/FDC Cross-Reference Header Record. The Cross-Reference Header shall be structured as follows:

Record Keyword Field (always 'RT')
Number of FID/FDC Cross-References Field

5.1.2.3.2.8.3 FID/FDC Cross-Reference Table Entry Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'RE')
Feature Identification Code Field
Feature Description Field
Feature Descriptor Code Field

5.1.2.3.2.9 Global-Based Texture Reference Table File. This table shall be included if there are any user defined global-based texture references. The name of this file shall be "CULrxxxxx.GTR", where "r" is "M" for Merged Culture Data, "0" for LOD 0 Culture Data, "1" for LOD 1 Culture Data, "2" for LOD 2 Culture Data, "3" for LOD 3 Culture Data, "4" for LOD 4 Culture Data, and "5" for LOD 5 Culture Data; and "xxxxx" is the culture tile sequence number. The Global-Based Texture Reference Table File format shall be as follows:

SIF File Identifier Record
Global-Based Texture Reference Table Header Record
for each Global-Based Texture Reference Table Entry
  Global-Based Texture Reference Record

5.1.2.3.2.9.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI CULTURE')
File Identifier Field (always 'GLOBAL-BASED TEXTURE REFERENCE TABLE')

5.1.2.3.2.9.2 Global-Based Texture Reference Table Header Record. The Global-Based Texture Reference Table Header shall be structured as follows:

Record Keyword Field (always 'GX')
Number of Texture References Field

5.1.2.3.2.9.3 Global-Based Texture Reference Record. The field structure shall be as follows:

Record Keyword Field (always 'GB')
Texture Reference Table Index Field
Texture Library Field
Texture ID Field
Texture Origin Field
Boundary ID Field
Mirror Field
Wrap Field
Wrap Type Field
Texture Scale Field
Orientation Vectors Field
Global Reference Point Field
Layer Number Field

5.1.2.3.2.10 Non-Mapped Texture Reference Table File. This table shall be included if there are any user defined non-mapped texture references. The name of this file shall be "CULrxxxxx.NTR", where "r" is "M" for Merged Culture Data, "0" for LOD 0 Culture Data, "1" for LOD 1 Culture Data, "2" for LOD 2 Culture Data, "3" for LOD 3 Culture Data, "4" for LOD 4 Culture Data, and "5" for LOD 5 Culture Data; and "xxxxx"is the culture tile sequence number. The Non-Mapped Texture Reference Table File format shall be as follows:

SIF File Identifier Record
Non-Mapped Texture Reference Table Header Record
for each Non-Mapped Texture Reference Table Entry
  Non-Mapped Texture Reference Record

5.1.2.3.2.10.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI CULTURE')
File Identifier Field (always 'NON-MAPPED TEXTURE REFERENCE TABLE')

5.1.2.3.2.10.2 Non-Mapped Texture Reference Table Header Record. The Non-Mapped Texture Reference Table Header shall be structured as follows:

Record Keyword Field (always 'NX')
Number of Texture References Field

5.1.2.3.2.10.3 Non-Mapped Texture Reference Record. The field structure shall be as follows:

Record Keyword Field (always 'NM')
Texture Reference Table Index Field
Texture Library Field
Texture ID Field

5.1.2.3.2.11 Model Reference Table File. This table shall be included if there are any model references. The name of this file shall be "CULrxxxxx.MRF", where "r" is "M" for Merged Culture Data, "0" for LOD 0 Culture Data, "1" for LOD 1 Culture Data, "2" for LOD 2 Culture Data, "3" for LOD 3 Culture Data, "4" for LOD 4 Culture Data, and "5" for LOD 5 Culture Data; and "xxxxx" is the culture tile sequence number. The Model Reference Table File format shall be as follows:

SIF File Identifier Record
Model Reference Header Record
for each Model Reference Table Entry
  Model Reference Table Entry Record

5.1.2.3.2.11.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI CULTURE')
File Identifier Field (always 'MODEL REFERENCE TABLE')

5.1.2.3.2.11.2 Model Reference Header Record. The Model Reference Header shall be structured as follows:

Record Keyword Field (always 'MR')
Number of Model References Field

5.1.2.3.2.11.3 Model Reference Table Entry Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'ME')
Model Reference Table Index Field
Model Number Field
Model LOD Field
Orientation Angle Field
Correlation Priority Field
Model Lat Long Field
Scale Factor Field
Model Library Type Field
Number of Substituted Features Field
for each Substituted Feature
    Substituted Feature Number Field

5.1.2.3.2.12 Superfeature File. This file shall be included if there are any superfeatures defined within the culture tile. The name of this file shall be "CULrxxxxx.SFR", where "r" is "M" for Merged Culture Data, "0" for LOD 0 Culture Data, "1" for LOD 1 Culture Data, "2" for LOD 2 Culture Data, "3" for LOD 3 Culture Data, "4" for LOD 4 Culture Data, or "5" for LOD 5 Culture Data; and "xxxxx" is the culture tile sequence number. The Superfeature file format shall be as follows:

SIF File Identifier Record
for each Superfeature
    Superfeature Header Record

5.1.2.3.2.12.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI CULTURE')
File Identifier Field (always 'SUPERFEATURE FILE')

5.1.2.3.2.12.2 Superfeature Header Record. There shall be a Superfeature header record for each superfeature defined within the culture tile. The field structure of this record shall be as follows:

Record Keyword Field (always 'SF')
Superfeature ID Field
Superfeature Description Field
Bounding Rectangle Coordinates Field
Number of Aggregate Features Field
Number of Child Features Field
Number of Child Superfeatures Field (currently 0 for P2851 SSDB data)
Number of Parent Superfeatures Field (currently 0 for P2851 SSDB data)
for each Aggregate Feature
    Feature Number Field
for each Child Feature
    Feature Number Field
for each Child Superfeature (currently none for P2851 SSDB data)
    Superfeature ID Field
for each Parent Superfeature (currently none for P2851 SSDB data)
    Superfeature ID Field

5.1.2.3.2.13 Feature File. The name of this file shall be "CULrxxxxx.FTR", where "r" is "M" for Merged Culture Data, "0" for LOD 0 Culture Data, "1" for LOD 1 Culture Data, "2" for LOD 2 Culture Data, "3" for LOD 3 Culture Data, "4" for LOD 4 Culture Data, and "5" for LOD 5 Culture Data; and "xxxxx" is the culture tile sequence number. The Feature File format shall be as follows, and as shown in Figure 13. Each title shall include a background areal feature.

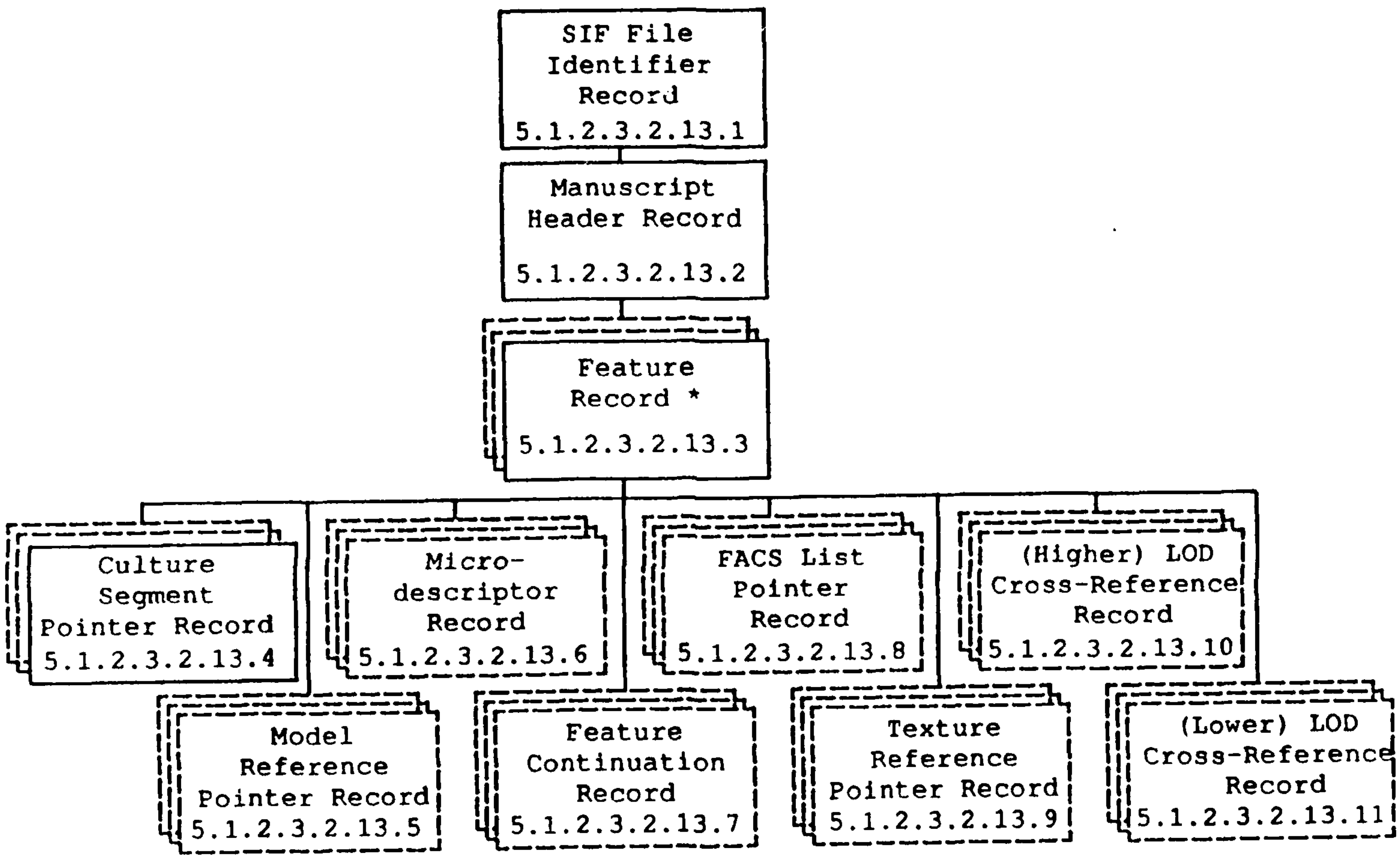
SIF File Identifier Record
Manuscript Header Record
for each Feature in the Feature File
  Feature Record
  for each Culture Segment Pointer
    Culture Segment Pointer Record
  for each Model Reference [optional]
    Model Reference Pointer Record
  for each Microdescriptor Record [optional]
    Microdescriptor Record
  for each Feature Continuation Record [optional]
    Feature Continuation Record
  for each FACS List Pointer [optional]
    FACS List Pointer Record
  for each Texture Reference Record [optional]
    Texture Reference Pointer Record
  for each Higher LOD Cross Reference [optional]
    LOD Cross Reference Record
  for each Lower LOD Cross Reference [optional]
    LOD Cross Reference Record

5.1.2.3.2.13.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI CULTURE')
File Identifier Field (always 'FEATURE FILE')

5.1.2.3.2.13.2 Manuscript Header Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'MH')
Highest Feature Number Field
Highest Segment Number Field
Cell Boundary Field
Manuscript Boundary Field
Security Level Field
Number of Areal Features Field
Number of Linear Features Field
Number of Point Features Field
Number of Point Light Features Field
Number of Point Light Strings Field
Number of Model References Field
Number of Texture References Field
Maintenance Date Field



*NOTE: A feature record may be an Areal, Linear, Point, Point Light, or a Point Light String Feature.

Figure 13. SIF/HDI Feature File's Record Relationships.

5.1.2.3.2.13.3 Feature Record. The Feature Record shall be one of the following record types:

Areal Feature Record
Linear Feature Record [optional]
Point Feature Record [optional]
Point Light Feature Record [optional]
Point Light String Feature Record [optional]

5.1.2.3.2.13.3.1 Areal Feature Record. The number of Areal Feature records shall correspond to the value in the Number of Areal Features field within the Manuscript Header Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'AF')
Feature Fragment Flag Field
Bounding Rectangle Coordinates Field
Number of Texture References Field
Number of Culture Segments Field
Number of FACS List Pointers Field
Number of Microdescriptors Field
Number of Instances Field
Number of Feature Continuations Field
Feature Number Field
Feature Identification Code Field
Feature Descriptor Code Field
Synthetic Data Flag Field
Source ID Number Field
Correlation Priority Field
Predominant Height Field
Surface Material Category Field
Color Table Index Field
Layer Number Field
Terrain Feature Identifier Field
Number of Higher LOD Cross References Field
Number of Lower LOD Cross References Field

5.1.2.3.2.13.3.2 Linear Feature Record. The number of Linear Feature records shall correspond to the value in the Number of Linear Features field within the Manuscript Header Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'LF')
Width Field
Bounding Rectangle Coordinates Field
Number of Texture References Field
Number of Culture Segments Field
Number of FACS List Pointers Field
Number of Microdescriptors Field
Number of Instances Field
Number of Feature Continuations Field
Feature Number Field
Feature Identification Code Field
Feature Descriptor Code Field
Synthetic Data Flag Field
Source ID Number Field
Correlation Priority Field
Predominant Height Field
Surface Material Category Field
Color Table Index Field
Layer Number Field
Terrain Feature Identifier Field
Number of Higher LOD Cross References Field
Number of Lower LOD Cross References Field

5.1.2.3.2.13.3.3 Point Feature Record. The number of Point Feature records shall correspond to the value in the Number of Point Features field within the Manuscript Header Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'PF')
Length Field
Orientation Field
Width Field
Bounding Rectangle Coordinates Field
Number of Texture References Field
Number of Culture Segments Field
Number of FACS List Pointers Field
Number of Microdescriptors Field
Number of Instances Field
Number of Feature Continuations Field
Feature Number Field
Feature Identification Code Field
Feature Descriptor Code Field
Synthetic Data Flag Field
Source ID Number Field
Correlation Priority Field
Predominant Height Field
Surface Material Category Field
Color Table Index Field
Layer Number Field
Terrain Feature Identifier Field
Number of Higher LOD Cross References Field
Number of Lower LOD Cross References Field

5.1.2.3.2.13.3.4 Point Light Feature Record. The number of Point Light Feature records shall correspond to the value in the Number of Point Light Features field within the Manuscript Header Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'PL')
Length Field
Orientation Field
Shape Code Field
Width Field
Directionality Field
Light Type Field
Number of Culture Segments Field
Number of FACS List Pointers Field
Number of Microdescriptors Field
Number of Instances Field
Number of Feature Continuations Field
Feature Number Field
Feature Identification Code Field
Feature Descriptor Code Field
Synthetic Data Flag Field
Source ID Number Field
Correlation Priority Field
Predominant Height Field
Surface Material Category Field
Color Table Index Field
Layer Number Field
Terrain Feature Identifier Field
Number of Higher LOD Cross References Field
Number of Lower LOD Cross References Field

5.1.2.3.2.13.3.5 Point Light String Feature Record. The number of Point Light String Feature records shall correspond to the value in the Number of Point Light Strings field within the Manuscript Header Record. The field structure of this record shall be as follows:

Record Keyword (always 'LS')
Length Field
Orientation Field
Width Field
Directionality Field
Light Type Field
Light String Shape Field
Number of Lights Field
Point Light String Origin Field
Point Light String Delta Field
Bounding Rectangle Coordinates Field
Number of Culture Segments Field
Number of FACS List Pointers Field
Number of Microdescriptors Field
Number of Instances Field
Number of Feature Continuations Field
Feature Number Field
Feature Identification Code Field
Feature Descriptor Code Field
Synthetic Data Flag Field
Source ID Number Field
Correlation Priority Field
Predominant Height Field
Surface Material Category Field
Color Table Index Field
Layer Number Field
Terrain Feature Identifier Field
Number of Higher LOD Cross References Field
Number of Lower LOD Cross References Field

5.1.2.3.2.13.4 Culture Segment Pointer Record. The total number of segment list pointers associated with a feature shall correspond to the value in the Number of Culture Segments Field in the parent Feature Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'SP')
Segment Direction Field
Correlation Priority Field
Segment ID Number Field

5.1.2.3.2.13.5 Model Reference Pointer Record. The number of these records shall correspond to the value in the Number of Instances field within the parent feature header record. The field structure of this record shall be as follows:

Record Keyword Field (always 'MP')
Model Reference Table Index Field

5.1.2.3.2.13.6 Microdescriptor Record. The total number of microdescriptors associated with a given feature shall correspond to the value in the Number of Microdescriptors field in the parent feature record. The field structure of this record shall be as follows:

Record Keyword Field (always 'MI')
Microdescriptor Type Field
Microdescriptor Value Field

5.1.2.3.2.13.7 Feature Continuation Record. The number of these records for a given feature shall correspond to the value in the Number of Feature Continuations field in the parent feature record. The field structure of this record shall be as follows:

Record Keyword Field (always 'FC')
Next Manuscript ID Field
Next Feature Number Field

5.1.2.3.2.13.8 FACS List Pointer Record. The total number of FACS list pointers associated with a feature shall correspond to the value in the Number of FACS List Pointers Field in the parent feature record. The field structure of this record shall be as follows:

Record Keyword Field (always 'FP')
FACS Table Index Field

5.1.2.3.2.13.9 Texture Reference Pointer Record. The total number of texture references associated with a feature shall correspond to the value in the Number of Texture References field within the parent feature record. The field structure of this record shall be as follows:

Record Keyword Field (always 'TX')
Texture Mapping Type Field
Texture Reference Table Index Field
Texture Mapping Set ID Field
Linear Feature Texture Orientation Field

5.1.2.3.2.13.10 Higher LOD Cross Reference Record. When multiple LODs are included, pointers shall be set between the coarse feature and its associated higher resolution feature(s). These pointers shall maintain a one lower LOD feature to many higher LOD feature relationship. The number of these records shall correspond to the Number of Higher LOD Cross References field in the parent feature record. The field structure of this record shall be as follows:

Record Keyword Field (always 'HL')
Next Manuscript ID Field
Next Feature Number Field

5.1.2.3.2.13.11 Lower LOD Cross Reference Record. A feature at a higher LOD shall incorporate a pointer to each lower LOD representation of that feature. The number of these records shall correspond to the Number of Lower LOD Cross References field in the parent feature record. The field structure of this record shall be as follows:

Record Keyword Field (always 'LL')
Next Manuscript ID Field
Next Feature Number Field

5.1.2.3.2.14 Segment File. The name of this file shall be "CULrxxxxx.SEG", where "r" is "M" for Merged Culture Data, "0" for LOD 0 Culture Data, "1" for LOD 1 Culture Data, "2" for LOD 2 Culture Data, "3" for LOD 3 Culture Data, "4" for LOD 4 Culture Data, and "5" for LOD 5 Culture Data; and "xxxxx" is the culture tile sequence number. The Segment File format shall be as follows, and as shown in Figure 14.

SIF File Identifier Record
for each Segment
  Segment Header Record
  Vertex List Pointer Record
  Segment Backpointer Record

5.1.2.3.2.14.1 SIF File Identifier Record. The field structure of this record shall be as follows:

Section Identifier Field (always 'SIF/HDI CULTURE')
File Identifier Field (always 'SEGMENT FILE')

5.1.2.3.2.14.2 Segment Header Record. There shall be a Segment Header Record for each coordinate segment defined within the culture file. The field structure of this record shall be as follows:

Record Keyword Field (always 'SH')
Segment ID Number Field
Beginning Coordinates Field
Ending Coordinates Field
Shared Segment Flag Field
Correlation Priority Field
Bounding Rectangle Coordinates Field
Clipped Boundary Flag Field
2-D/3-D Coordinates Flag Field
Number of Coordinate Pointers Field
Number of Segment Backpointers Field

5.1.2.3.2.14.3 Vertex List Pointer Record. The number of entries in this array shall correspond to the Number of Coordinate Pointers Field in the parent Segment Header Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'VP')
for each Vertex List Pointer
  Vertex Pointer Field

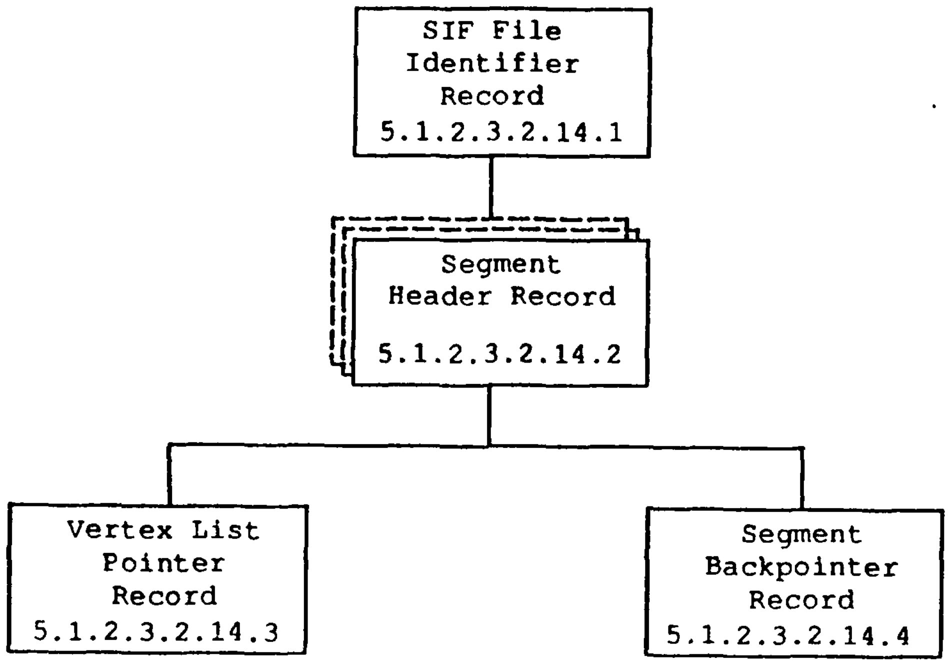


Figure 14. <u>SIF/HDI Segment File's Record Relationships</u>.

5.1.2.3.2.14.4 Segment Backpointer Record. The total number of segment backpointers associated with a segment shall correspond to the value in the Number of Segment Backpointers Field in the parent Segment Header Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'SB')
for each backpointer
Feature Number Field

5.1.2.4 Gridded Data

5.1.2.4.1 Gridded Data Encoding. The National Imagery Transmission Format (NITF) version 1.1, shall be used to store SIF/HDI rasterized photo texture, terrain elevation data (where elevation is substituted for image pixel values) and sensor texture data (where feature identifiers and surface material categories are substituted). Extensions to the published NITF format, needed to support SIF/HDI, shall be limited to those documented in this standard. A SIF/HDI data set shall support multiple terrain levels of detail (LODs) expressed in fixed units of latitude and longitude, spaced at 3, 1, 0.3, and 0.03 arc seconds.

50.1.2.4.2 Gridded Data Section Structure

5.1.2.4.2.1 Basic NITF structure. As defined in the NITF documentation (NITF, version 1.1, CN No. 2) each header and sub-header shall have its own file, which shall be in ASCII. All sizes for data fields are in bytes, with one byte per character. Each grid shall be stored in its own binary file. These files shall contain field labels on odd-numbered lines and field values (corresponding to the immediately preceding field label) on even-numbered lines. Each field label line shall be in all capital letters, and may include numerics. Each line shall be terminated with a Carriage Return/Line Feed pair. The file shall be terminated with an end-of-file (CNTRL-Z) character.

5.1.2.4.2.2 SIF/HDI application-specific features

5.1.2.4.2.2.1 Data Fill. The following rules shall be used to populate data fields. Each field is fixed-length.

a. Data in text fields shall be left justified and padded with blanks.

b. Data in numeric integer fields shall be right justified and padded with leading zeroes.

c. Data in numeric floating point fields shall be right justified and padded with leading blanks.

d. Numeric floating point fields shall be provided in scientific notation as defined in the SIF/HDI and SIF/DP Data Dictionary.

e. Null values shall be blanks for all fields.

5.1.2.4.2.2.2 Field Types. The NITF standard defines three types of fields for headers and sub-headers: Required (R), Optional (O), and Conditional (C). A Required field shall be present and must contain valid data. An Optional field shall be present, but may or may not contain valid data. If there is no valid data for the field, it shall be blank-filled. A Conditional field may or may not be present depending on the value of preceding field(s); if it is present, then it shall contain valid data. This standard defines an additional field type: Null (N). A Null field shall be present, and shall contain a series of blanks filling the field. For user-defined sub-headers, the field type shall be based on the texture library type of the image.

5.1.2.4.2.2.3 Data Classes. SIF/HDI shall use only the NITF image data class. The image class shall be used to handle general texture, which could be generic, geospecific, or model-specific. The symbol, label, text, audio, and non-static presentation information (NPI) data classes shall not be used in SIF/HDI.

5.1.2.4.2.2.4 File Order. All terrain grid files shall appear first, followed by all texture grid files. The SIF/HDI gridded data section format shall be as follows, and as shown in Figure 15.

```
NITF Header File
for each terrain tile
      Terrain Sub-Header File
      Terrain Data File
for each texture tile
      Image Sub-Header File
      Image Data File
```

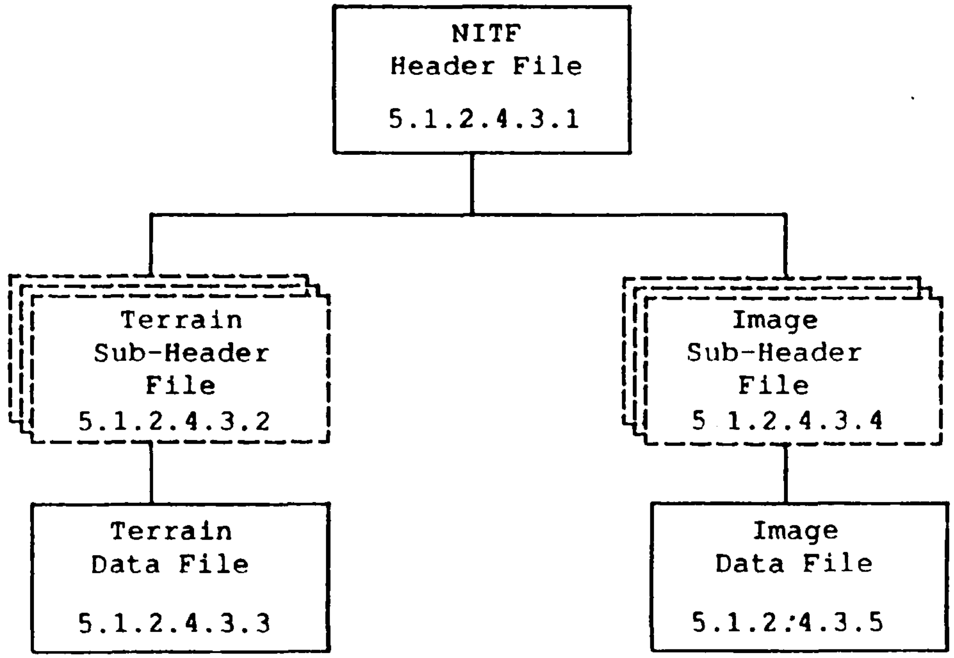


Figure 15. SIF/HDI Gridded Data File's Relationships.

5.1.2.4.3 Gridded Data File Structures

5.1.2.4.3.1 NITF Header File. The NITF Header format shall be in accordance with version 1.1 of NITF. All fields in the NITF Header shall be encoded as ASCII characters, including numeric values. The name of the file shall be "NITF.HDR". The file format shall be as follows:

Label	Field
MHDR	Message Type & Version
STYPE	System Type
OSTAID	Originating Station ID
MDT	Message Date & Time
MTITLE	Message Title
MSCLAS	Message Security Classification
MSCODE	Message Codewords
MSCTLH	Message Control and Handling
MSREL	Message Releasing Instructions
MSCAUT	Message Classification Authority
MSCTLN	Message Security Control Number
MSDWNG	Message Security Downgrade
MSDEVT	Message Downgrading Event
MSCOP	Message Copy Number
MSCPYS	Message Number of Copies
ENCRYP	Encryption
ONAME	Originator's Name
OPHONE	Originator's Phone Number
ML	Message Length
HL	NITF Header Length
NUMI	Number of Images
	for each image
LISHnnn	Length of Image Sub-Header
LInnn	Length of Image
NUMS	Number of Symbols [always zero]
NUML	Number of Labels [always zero]
NUMT	Number of Text Files [always zero]
NUMA	Number of Audio Segments [always zero]
NUMF	Number of Non-Static Presentation Information Files [always zero]
UDHDL	User Defined Header Data Length
	SIF/HDI User Defined Header Data (subrecord)
XHDL	Extended Header Data Length [always zero]
XHD	Extended Header Data [always zero]

5.1.2.4.3.1.1 SIF/HDI User Defined Header Data. Texture data shall be treated as imagery, and be counted within the Number of Images field in the basic NITF Header. The field structure of this subrecord shall be as follows.

```
     Label      Field

     UDHD       Data Base Sentinel (always "SIF/HDI")


TERRAIN DATA:
     NUMTER     Number of Terrain Files
                for each terrain file
     TERSHL          Length of Terrain Sub-Header File
     TERFL           Length of Terrain File


IMAGE TIE POINT DATA:
     NUMGTP     Number of Geographic Tie Points
                for each geographic (areal) tie point
     GTPID           Geographic Tie Point ID
     NUMGTPR         Number of Tie Point References
                     for each tie point reference
     STEXLIB              Texture Library (Stage 1 or 2 Areal only)
     TEXID                Texture ID
     NUMMTP     Number of Model Tie Points
                for each model tie point
     MTPID           Model Tie Point ID
     NUMMTPR         Number of Tie Point References
                     for each tie point reference
     STEXLIB              Texture Library (Stage 1 or 2 Model only)
     TEXID                Texture ID


GENERIC TEXTURE ASSOCIATION DATA:
     NUMGTS     Number of Generic Texture Sets
                for each generic texture set
     GTSNAME         Generic Texture Set Name
     OMTF            Object Or Material Texture Flag
     NUMGT           Number of Generic Textures In Set
                     for each generic texture
     TEXID                Texture ID
```

5.1.2.4.3.2 Terrain Sub-Header File. Unlike NITF, the image size shall have no logical limitation. All SIF/HDI terrain elevation data shall be 24 bits in length. Corner coordinates shall be expressed in units of thousandths of arc seconds. Terrain shall be stored in the WGS-84 geodetic coordinate system. Within a terrain tile, there may be rectangular insets whose source is different from that of the rest of the tile. These insets shall always be at the same resolution and spacing as the rest of that tile. Within the SIF/HDI User-Defined Terrain Data, the first source listed shall always be primary source (i.e., the source for the background terrain data). All subsequent sources shall be for the insets whose boundaries are defined in the Terrain Source Footprint Data section in the SIF/HDI User-Defined Terrain Data. A single boundary shall indicate that there are no insets, and there is only a single source. This shall be the usual case. All boundaries shall be rectangular and shall be oriented in a North-South, East-West orientation. The name of this file shall be "TERxxxxx.HDR", where "xxxxx" is the terrain tile sequence number.

Label	Field
TM	Message Part Type [always "TM"]
TID	Terrain ID
TDATIM	Terrain Date & Time
TGTID	Target ID
TTITLE	Terrain Title
TSCLAS	Terrain Security Classification
TSCODE	Terrain Codewords
TSCTLH	Terrain Control and Handling
TSREL	Terrain Releasing Instructions
TSCAUT	Terrain Classification Authority
TSCTLN	Terrain Security Control Number
TSDWNG	Terrain Security Downgrade
TSDEVT	Terrain Downgrading Event
ENCRYP	Encryption
TSORCE	Terrain Source
TCORDS	Terrain Coordinate System [always geodetic/geographic]
TGEOLO	Terrain Geographic Location
NTCOM	Number of Terrain Comments
	for each Terrain Comment n
TCOMn	Terrain Comment
TC	Terrain Compression
COMRAT	Compression Rate Code
NBANDS	Number of Bands [always 1 for terrain]
	for each band n
TTYPEn	Terrain Type
TFCn	Terrain Filter Condition
TEFLTn	Standard Terrain Filter Code
NLUTSn	Number of LUTs [always 0 for terrain]
TSYNC	Terrain Sync Code
TMODE	Terrain Mode [always band sequential]
NBPR	Number of Blocks Per Row
NBPC	Number of Blocks Per Column
NPPBH	Number of Pixels Per Block Horizontal
NPPBV	Number of Pixels Per Block Vertical
NBPP	Number of Bits Per Pixel Per Band
DLVL	Display Level
ALVL	Attachment Level
TLOC	Terrain Location
TMAG	Terrain Magnification

5.1.2.4.3.2 Terrain Sub-Header File - Continued.

Label	Field
.	
UDTDL	User Defined Terrain Data Length
	User Defined Terrain Data
XSHDL	Extended Sub-Header Data Length
XSHD	Extended Sub-Header Data [always zero]

5.1.2.4.3.2.1 SIF/HDI User Defined Terrain Data. The field structure of this subrecord shall be as follows.

Label	Field
UDTD	SIF/HDI Sentinel (always "SIF/HDI")

GENERAL PROCESSING DATA:

HRES	Horizontal Resolution
VRES	Vertical Resolution
HSIZE	Horizontal Size
VSIZE	Vertical Size
ODS	Original Data Sources Used
PAST	Positional Accuracy Standards
EAST	Elevation Accuracy Standards
ELRES	Elevation Resolution

SOURCE DATA:

NUMDS	Number of Data Sources
	for each data source
SOID	Source ID Number
SOTYPE	Source Type
SONAME	Source Name
͜OAP	Source Agency/Project
SODATE	Source Date
REDA	Reliability of Data
COLSYS	Collection System
CODATE	Compilation Date
SYNDF	Synthetic Data Flag
COMCRI	Compilation Criteria

TERRAIN SOURCE FOOTPRINT DATA:

NUMBOU	Number of Boundaries
	for each boundary
BOUNDID	Boundary ID
SOID	Source ID Number
NUMBP	Number of Boundary Points
	for each boundary point
BPID	Boundary Point ID
LATLON	Boundary Coordinates

5.1.2.4.3.3 Terrain Data File. The name of this file shall be"TERxxxxx.DAT", where "xxxxx" is the terrain tile sequence number. The field structure of this file shall be as follows. Elevation values shall be given in binary integer form as defined by NITF.

```
for each row from top to bottom
      for each column from left to right
            elevation value
```

5.1.2.4.3.4 Image Sub-Header File. The SIF/HDI Image Coordinate System shall be geodetic/geographic unless an image is being transmitted in its original format which is not in geodetic/geographic coordinates. Corner coordinates shall be expressed in units of thousandths of arc seconds. The image size shall have no logical limitation. The SIF/HDI implementation of NITF shall not use Look Up Tables (LUTs) for visual (color or intensity) texture. All such data shall be directly stored in the NITF Image Data File. For SMC/FDC data, LUTs may or may not be used. If LUTs are used, the LUT entry shall be entirely in ASCII with a length of seven bytes. The first two bytes shall represent the SMC (0 - 15), while the following five bytes shall represent the ASCII FDC value. The name of this file shall be "TEXxxxxx.HDR", where "xxxxx" is the texture tile sequence number.

Label	Field
IM	Message Part Type [always "IM"]
IID	Image ID (unique across SIF database)
IDATIM	Image Date & Time
TGTID	Target ID
ITITLE	Image Title
ISCLAS	Image Security Classification
ISCODE	Image Codewords
ISCTLH	Image Control and Handling
ISREL	Image Releasing Instructions
ISCAUT	Image Classification Authority
ISCTLN	Image Security Control Number
ISDWNG	Image Security Downgrade
ISDEVT	Image Downgrading Event
ENCRYP	Encryption
ISORCE	Image Source
ICORDS	Image Coordinate System
IGEOLO	Image Geographic Location
NICOM	Number of Image Comments
	for each Image Comment n
ICOMn	Image Comment
IC	Image Compression
COMRAT	Compression Rate Code
NBANDS	Number of Bands
	for each band n
ITYPEn	Image Type
IFCn	Image Filter Condition
IMFLTn	Standard Image Filter Code
NLUTSn	Number of LUTs [SIF/HDI defaults to 0]
	for each LUT m
NELUTm	Number of LUT Entries
	for each LUT entry e
LUTDe	LUT Entry Data
ISYNC	Image Sync Code
IMODE	Image Mode [SIF/HDI defaults to Band Sequential]
NBPR	Number of Blocks Per Row
NBPC	Number of Blocks Per Column
NPPBH	Number of Pixels Per Block Horizontal [SIF/HDI default = 64]
NPPBV	Number of Pixels Per Block Vertical [SIF/HDI default = 64]
NBPP	Number of Bits Per Pixel Per Band
DLVL	Display Level
ALVL	Attachment Level

5.1.2.4.3.4 Image Sub-Header File - Continued

Label	Field
ILOC	Image Location
IMAG	Image Magnification
UDIDL	User Defined Image Data Length
	SIF/HDI User Defined Image Data (subrecord)
XSHDL	Extended Sub-Header Data Length
XSHD	Extended Sub-Header Data [reserved]

5.1.2.4.3.4.1 SIF/HDI User Defined Image Data. SIF/HDI shall include any or all of the following types of texture.

5.1.2.4.3.4.1.1 Stage 1 Specific Areal. Stage 1 Specific Areal Texture (A1) shall consist of images whose contents have not been changed through any geometric or radiometric operations. All such images shall be exchanged in the NITF format specified in this section. Ground control points shall be provided with these images. Tie points shall be provided.

5.1.2.4.3.4.1.2 Stage 2 Specific Areal. Stage 2 Specific Areal Texture (A2) shall consist of images whose contents have been changed through radiometric and cut/paste operations only. Such operations shall include noise removal, occlusion removal, shadow minimization, haze removal, and color and contrast enhancements. The auxiliary information provided with Stage 1 Specific Areal Textures (ground control points and tie points) shall also be included.

5.1.2.4.3.4.1.3 Stage 3 Specific Areal. Stage 3 Specific Areal Texture (A3) shall consist of images whose contents have been changed through radiometric, cut/paste, and geometric operations. Such operations shall include noise removal, occlusion removal, shadow minimization, haze removal, color and contrast enhancements, image-to-image contrast enhancement, and orthorectification to include both 2D geometric corrections and 3D geometric corrections. These images shall be in the geodetic coordinate system with equal arc spacing.

5.1.2.4.3.4.1.4 Stage 1 Specific Model. Stage 1 Specific Model Texture (M1) shall consist of images whose contents have not been changed through any kind of geometric or radiometric operations. All such images shall be exchanged in the NITF format specified in this section. Control points in the model's local coordinate system shall be provided. Tie points shall be provided with these images. These images shall be in the local cartesian coordinate system in units of meters.

5.1.2.4.3.4.1.5 Stage 2 Specific Model. Stage 2 Specific Model Texture (M2) shall consist of images whose contents have been changed through radiometric and cut/paste operations only. Such operations shall include noise removal, occlusion removal, shadow minimization, haze removal, and color and contrast enhancements. Control points and tie points shall also be included. These images shall be in the local cartesian coordinate system in units of meters.

5.1.2.4.3.4.1.6 Stage 3 Specific Model. Stage 3 Specific Model Texture (M3) shall consist of images whose contents have been changed through radiometric, cut/paste, and geometric operations. Such operations shall include noise removal, occlusion removal, shadow minimization, haze removal, color and contrast enhancements, image-to-image contrast enhancement, and orthorectification to include both 2D geometric corrections and 3D geometric corrections. These images shall be in the local cartesian coordinate system in units of meters.

5.1.2.4.3.4.1.7 Generic. Generic Texture (G) shall consist of non-geospecific images, for both geographic areas as well as models. Such texture shall be radiometrically and geometrically corrected. Generic texture shall be stored in units of meters.

5.1.2.4.3.4.1.8 SMC/FDC. SMC/FDC Texture (SF) shall consist of the DMA Surface Material Category (SMC) and Feature Descriptor Code (FDC) for that position. Such texture shall be geometrically corrected. These images shall be in the geodetic coordinate system with equal arc spacing. SMC/FDC texture shall be provided for geospecific areas only.

5.1.2.4.3.4.2 Field Structure. The field structure of this subrecord shall be as follows. For each texture type, a field shall be Required (R), Optional (O), Conditional (C), or Null (N), as specified herein.

Label	Field	A 123	M 123	G	S F
UDID	Data Base Sentinel (always"SIF/HDI")	RRR	RRR	R	R
GENERAL PROCESSING DATA:					
STEXLIB	Texture Library	RRR	RRR	R	R
TEXID	Texture ID (unique within a Texture Lib.)	RRR	RRR	R	R
STEXID	SSDB Texture ID (original SSDB Tex. ID)	RRR	RRR	R	R
TTYPE	Texture Type	RRR	RRR	R	R
TEXDES	Texture Description	RRR	RRR	R	R
HRES	Horizontal Resolution	RRR	RRR	R	R
VRES	Vertical Resolution	RRR	RRR	R	R
HSIZE	Horizontal Size	RRR	RRR	R	R
VSIZE	Vertical Size	RRR	RRR	R	R
MSTF	Modified Specific Texture Flag	RRR	RRR	N	O
NRF	Noise Removal Flag	RRR	RRR	N	O
ORF	Occlusion Removal Flag	RRR	RRR	N	O
HRF	Haze Removal Flag	RRR	RRR	N	N
SMF	Shadow Minimization Flag	RRR	RRR	N	N
IICEF	Inner Image Contrast Enhancement Flag	RRR	RRR	N	N
ITICEF	Image-to-Image Contrast Enhancement Flag	RRR	RRR	N	N
2GCF	2-D Geometric Correction Flag	RRR	RRR	N	R
3GCF	3-D Geometric Correction Flag	RRR	RRR	N	R
PRCOM	Processing Comments	OOO	OOO	O	O
IQC	Image Quality Comment	OOO	OOO	O	O
IQR	Image Quality Rating	OOO	OOO	O	O
ICAPDT	Image Capture Date & Time	RRR	RRR	O	R
IFCRDT	Image File Creation Date & Time	OOO	OOO	O	O
LMDT	Last Maintenance Date & Time	RRR	RRR	R	R
PAST	Positional Accuracy Standards	OOO	OOO	O	O
GEOLOC	Geographic Location Name	RRR	OOO	O	R
GTSNAME	Generic Texture Set Name	NNN	NNN	R	N

5.1.2.4.3.4.2 Field Structure - Continued.

Label	Field	A 123	M 123	G	S F
SOURCE DATA:					
NUMDS	Number of Data Sources	RRR	RRR	R	R
	for each data source				
SOID	Source ID Number	RRR	RRR	R	R
SOTYPE	Source Type	RRR	RRR	R	R
SONAME	Source Name	RRR	RRR	R	R
SOAP	Source Agency/Project	RRR	RRR	R	R
SODATE	Source Date	RRR	RRR	R	R
SEID	Sensor ID	RRO	RRO	N	O
SETYPE	Sensor Type	RRO	RRO	N	O
SENAME	Sensor Name	RRO	RRO	N	O
REDA	Reliability of Data	RRR	RRR	R	R
PAST	Positional Accuracy Standards	OOO	OOO	O	O
COLSYS	Collection System	RRR	RRR	R	R
CODATE	Compilation Date	RRR	RRR	R	R
SYNDF	Synthetic Data Flag	RRR	RRR	R	R
COMCRI	Compilation Criteria	OOO	OOO	O	O
ICAPDT	Image Capture Date & Time	RRR	RRR	O	R
ENVIRONMENTAL CONDITIONS DATA:					
SPENVC	Special Environmental Conditions	OOO	OOO	N	O
PERCC	Percent of Cloud Cover	RRO	OOO	N	N
PERSC	Percent of Shadow Cover	RRO	OOO	N	N
TEXTURE FOOTPRINT DATA:					
PERTT	Percent of Texture in Tile	RRR	RRR	R	R
PERST	Percent of Specific Texture	RRR	RRR	O	R
NUMBOU	Number of Boundaries	RRR	RRR	R	R
	for each boundary				
BOUNDID	Boundary ID	CCC	CCC	C	C
SOID	Source ID Number	CCC	CCC	C	C
MODVIEW	Model View Description (Stage 3)	CCC	CCC	C	C
NUMBP	Number of Boundary Points	CCC	CCC	C	C
	for each boundary point				
BPID	Boundary Point ID	CCC	CCC	C	C
LATLON	Absolute Latitude/Longitude	CCC	NNN	N	C
RELCO	Relative Coordinates	NNN	CCC	N	N
ICO	Image Coordinates	CCC	CCC	C	C

5.1.2.4.3.4.2 Field Structure - Continued.

Label	Field	A 123	M 123	G	SF
NEIGHBOR TEXTURE ASSOCIATION DATA:					
NOTNID	North Tile Neighbor ID	NNO	NNN	N	O
SOTNID	South Tile Neighbor ID	NNO	NNN	N	O
EATNID	East Tile Neighbor ID	NNO	NNN	N	O
WETNID	West Tile Neighbor ID	NNO	NNN	N	O
ABTNID	Above Tile Neighbor ID	NNN	NNO	N	N
BETNID	Below Tile Neighbor ID	NNN	NNO	N	N
RITNID	Right Tile Neighbor ID	NNN	NNO	N	N
LETNID	Left Tile Neighbor ID	NNN	NNO	N	N
MODEL ASSOCIATION DATA:					
NUMMI	Number of Models in Image	NNN	RRR	N	N
	for each model				
MODLIB	Model Library Type	CCC	CCC	C	C
MODNUM	Model Number	CCC	CCC	C	C
MODNAME	Model Name	CCC	CCC	C	C
MODVIEW	Model View Description	CCC	CCC	C	C
IMAGE CONTROL DATA:					
NUMCP	Number of Control Points	RRO	RRO	N	O
	for each control point				
CPID	Control Point ID	RRC	RRC	N	C
CPNAME	Control Point Name	RRC	RRC	N	C
SOID	Source ID Number	RRC	RRC	N	C
LATLON	Absolute Latitude/Longitude	RRC	NNN	N	C
RELCO	Relative Coordinates	NNN	RRC	N	N
ICO	Image Coordinates	RRC	RRC	N	C
NUMGTP	Number of Geographic Tie Points	RRO	NNN	N	O
	for each geographic tie point reference				
GTPID	Geographic Tie Point ID	CCC	CCC	C	C
ICO	Image Coordinates	CCC	CCC	C	C
NUMMTP	Number of Model Tie Points	NNN	RRO	N	O
	for each model tie point reference				
MTPID	Model Tie Point ID	CCC	CCC	C	C
ICO	Image Coordinates	CCC	CCC	C	C
MODLIB	Model Library Type	CCC	CCC	C	C
MODNUM	Model Number	CCC	CCC	C	C

5.1.2.4.3.4.2 Field Structure - Continued.

Label	Field	A 123	M 123	G	S F
SENSOR IMAGE DESCRIPTOR DATA:					
NUMSEN	Number of Sensors	RRR	RRR	N	R
	for each sensor				
SEID	Sensor ID	RRO	OOO	C	O
FILMQ	Film Quality	OOO	OOO	C	O
SUNAZ	Sun Azimuth	RRO	OOO	C	O
SUNEL	Sun Elevation	RRO	OOO	C	O
NUMSTM	Number of Stereo Mates	RRO	OOO	C	O
	for each stereo mate				
TEXID	Texture ID	CCC	CCC	C	C
SCANID	Scanner ID	OOO	OOO	C	O
SCRES	Scan Resolution	OOO	OOO	C	O
SCFID	Scan Filter ID	OOO	OOO	C	O
LLCOR	LL Corner X/Y Image Coordinates	RRO	RRO	C	O
ULCOR	UL Corner X/Y Image Coordinates	RRO	RRO	C	O
URCOR	UR Corner X/Y Image Coordinates	RRO	RRO	C	O
LRCOR	LR Corner X/Y Image Coordinates	RRO	RRO	C	O
CALFL	Calibrated Focal Length	RRO	OOO	C	O
CALPPO	Calibrated Principal Point Offset	RRO	OOO	C	O
CALPSO	Calibrated Point of Symmetry Offset	RRO	OOO	C	O
NUMFID	Number of Fiducial Coordinates	RRO	RRO	C	O
	for each fiducial coordinate				
CALRIC	Calibration Report Image Coordinates	CCC	CCC	C	C
MEAIC	Measured Image Coordinates	CCC	CCC	C	C
OMEGA	Omega	RRO	OOO	C	O
PHI	Phi	RRO	OOO	C	O
KAPPA	Kappa	RRO	OOO	C	O
RECTIF	Rectification	RRO	OOO	C	O
CAMPLL	Camera Position in Lat/Lon	RRO	OOO	C	O
CAMPH	Camera Position in Height	RRO	RRO	C	O
MSEOPK	Mean Square Error Omega/Phi/Kappa	CCC	CCC	C	C
MSELLH	Mean Square Error Latitude/Longitude/Height	CCC	CCC	C	C
HCAPTS	Horizontal Captured Texel Size	RRO	RRO	C	O
VCAPTS	Vertical Captured Texel Size	RRO	RRO	C	O

5.1.2.4.3.5 Image Data File. The SIF/HDI implementation of the NITF standard shall store the actual band value(s) at each texel position (e.g., the red, green, and blue intensity values). It shall not use look-up tables (LUTs) except optionally for SMC/FDC data. As specified by NITF, each band shall have the same number of bits. The name of this file shall be "TEXxxxxx.DAT", where "xxxxx" is the texture tile sequence number.

5.1.2.4.3.5.1 SMC/FDC Encoding. A single SMC/FDC code shall require six bytes of storage, in one band of data. The high-order byte shall represent the SMC value in simple binary integer format. Valid SMC values range from 0 to 15. The following five bytes shall represent the FDC value in ASCII. The SMC/FDC values may be stored directly in the Image Data File; however, it is strongly recommended that an LUT be used for SMC/FDC values so that the Image Data File only stores indices into the LUT.

5.1.2.4.3.5.2 Grid Size. The grid size, as well as horizontal and vertical spacing of grid posts, shall conform to the technical characteristics of the SDBF data base system, as documented in the Project 2851 Software Design Document.

5.1.2.4.3.5.3 Data Compression. Although NITF allows several forms of data compression, compression shall only be applied to SIF/HDI using the lossless Joint Photographic Experts Group (JPEG) algorithm.

5.1.2.4.3.5.4 Record Structure. This file shall consist of a single logical record containing a simple byte stream. The field structure of this record shall be as follows. Texel values shall be in binary integer form beginning with the most significant bit (MSB) and ending with the least significant bit (LSB). The integer shall be interpreted as pure magnitude with no sign bit.

```
if Image Mode is Band Sequential (the SIF/HDI default)
     for each band
          for each block
               for each row from top to bottom
                    for each column from left to right
                         texel value

elseif Image Mode is Band Interleaved
     for each block
          for each band
               for each row from top to bottom
                    for each column from left to right
                         texel value
```

5.1.3 SIF/HDI Data Base Content. The subsequent paragraphs define the manner in which the SIF/HDI file format, specified in section 5.1.2, shall be populated.

5.1.3.1 SDBF Produced Data Sets. SIF/HDI data sets produced by the Simulator Data Base Facility will reflect the full information content of the SSDB, at the time the data set is created. The content of the SIF/HDI data set will be limited only by the selection of files to be output, and the geographic area of coverage specified. The levels of detail, layers, and resolutions of information within the SIF data set will correspond exactly to those of the SSDB. Non-optional data records and fields for which ground-truth information does not exist will be populated with default values. A SDBF-produced SIF data set will be classified at the level of the SSDB from which it was generated, typically SECRET/NOFORN (collateral).

5.1.3.2 Externally Produced Data Sets. A SIF/HDI data set generated by an external producer shall contain all of the information in the source data set from which it was translated. All values used to populate mandatory fields shall fall within permissible ranges, as defined within this standard.

5.1.3.2.1 Classified Information. SIF/HDI data sets may be classified at any level up to and including TOP SECRET. Data sets shall be generated under Special Access Required programs only when they can be released to the SDBF, and only when the information contained in them can be downgraded to a collateral level when incorporated in the SSDB.

5.1.3.2.2 Proprietary Information. SIF/HDI data sets shall not contain proprietary information. Proprietary information in the source data base shall be converted into a non-proprietary form during the conversion into SIF. When it appears necessary that information be modified or omitted from a SIF data set for proprietary reasons, the SIF data set shall only be generated with the knowledge and consent of the acquisition agency and the SDBF.

5.1.3.2.3 Mandatory Content. All data items which are not labeled "optional" within section 5.1.2 of this standard shall be considered mandatory, and thus populated by the producer. Mandatory items for which the producer does not have source information shall be populated with default or synthetic data. The producer shall indicate the presence of default or synthetic information by setting the corresponding flags in the appropriate records.

5.1.3.2.4 Optional Content. Data items labeled "optional" within this standard shall be populated as directed by the acquisition agency, or at the discretion of the producer, with Government concurrence.

5.1.3.2.4.1 Optional FACS Records. FACS records shall be defined for the representation of source data base information which does not directly correspond with any predefined SIF/HDI record. Contingent upon SDBF approval, these records shall be populated with the appropriate information during the generation of the SIF/HDI data set.

5.1.3.2.5 Data Quality. The data content of SIF data sets shall meet the quality criteria specified herein.

5.1.3.2.5.1 General. The following shall apply to all sections of the SIF data set.

5.1.3.2.5.1.1 Boundary Integrity. For any specified area of coverage, boundary integrity shall be maintained as follows:

a. There shall be no data with coordinates falling outside the boundary (cell, manuscript, or area block), as defined in the applicable header.

b. There shall be no gaps in data coverage over the specified area.

c. There shall be no redundant data coverage over the specified area.

5.1.3.2.5.1.2 Data Values. Data values shall be as defined in Appendix A of this standard.

5.1.3.2.5.1.3 Source Traceability. The sources used in the generation of all SIF data shall be identified. Traceability at the file level shall be required; optionally, sources shall be identified at the feature level, as well.

5.1.3.2.5.1.4 Levels of Detail. SIF data sets shall be segregated into multiple, correlated levels of detail whenever technically possible. The allocation of a specific model or feature to a particular level of detail shall follow the guidance established by the SDBF. Data at different resolutions or levels of detail covering a given area shall be fully correlated through population of the LOD pointers.

5.1.3.2.5.1.5 Post-Acceptance SIF Generation. When an external producer delivers a SIF data set as a by-product of the generation of a real-time trainer data base, the SIF data set shall be generated after the Government acceptance testing of the real-time data base.

5.1.3.2.5.2 Culture Data Section. The quality of feature data shall be quantified in terms of positioning accuracy, attribution accuracy, and conformance to capture criteria.

5.1.3.2.5.2.1 Capture Criteria. Feature capture criteria shall be observed to avoid the omission of expected features, presence of unexpected features, and improper aggregation of features. At the lower levels of detail, capture criteria conformance shall be measured relative to standard DLMS levels or map sheets.

5.1.3.2.5.2.2 Derivative Areal Features. The SIF data set shall not contain derivative areal features, i.e., those which have been decomposed into multiple polygons for the purpose of eliminating concavity and/or enforcing co-planarity with a polygonized terrain model.

5.1.3.2.5.2.3 Radar Characteristics. In SIF data sets created from radar simulation data bases, SIF producers shall provide the appropriate gamma curves, stored in User-Defined FACS tables, when available.

5.1.3.2.5.3 Gridded Data Section

5.1.3.2.5.3.1 Terrain Post Spacing. Terrain grid posts shall be located based upon a geodetic (arc-second) grid. An external producer's non-geodetic data base shall be resampled, such that the SIF data set generated from it contains a geodetic grid. The fact that this operation has been performed shall be recorded in the gridded data section of the SIF data set, as well as in the corresponding data base descriptive document.

5.2 SIF/DP Data Base Format

5.2.1 SIF/DP Data Base Structure

5.2.1.1 Logical Format. The logical format of a SIF/DP data base shall be made up of a hierarchy of data entities as follows:

Data Base

Section

File

Record

Field

Item

5.2.1.1.1 Data Base. The data base shall consist of a data base header file and all the requested models, culture, terrain, and texture for a specified geographic area. If any models are requested, then the entire model library shall be transmitted. Logically, the data base shall consist of a data base header file and one, two, three, or four sections.

5.2.1.1.2 Section. A section shall consist of a series of files containing information for a certain type of data: (1) models, (2) culture, (3) terrain, or (4) texture. Within a database, there shall be either one section or no sections for each of these four types.

5.2.1.1.3 File/Record/Field/Item. A file shall consist of a series of records, a record shall consist of a series of fields, and a field shall consist of one or more items. The item shall be the lowest logical data entity in the data base.

5.2.1.2 Physical Format. All files shall be stored within units known as "save sets" produced and read by the VAX/Virtual Memory System (VMS) Backup utility. One or more files may be contained within a single save set. The physical format of the SIF/DP data base shall be as follows.

5.2.1.2.1 Data Order. The physical order of data in the SIF/DP data base shall be as follows:

SIF/DP Data Base Header File Save Set
Model Data Section Save Set(s) [optional]
Culture Data Section Save Set(s) [optional]
Terrain Data Section Save Set(s) [optional]
Texture Data Section Save Set(s) [optional]

5.2.1.2.2 Physical Tape Format. The physical tape format of a SIF/DP data base shall be the VAX/VMS ANSI-labeled magnetic tape format, which adheres to Level 3 of the ANSI Standard for Magnetic Tape Labels and File Structure for Information Interchange, ANSI X3.27. The format of the physical tape shall be as follows:

Beginning-of-Tape Marker (BOT)

Volume Label (VOL1)

for each file (save set)
    File Header Labels (HDR1, HDR2)
    Tape Mark (TM)
    File Section
    Tape Mark (TM)
    File Trailer Labels (EOF1, EOF2 or EOV1, EOV2)

Tape Mark (TM)

Tape Mark (TM)

Scratch Tape

End-of-Tape Marker (EOT)

5.2.1.2.2.1 File Boundaries. An individual file may cross a tape boundary; in such a case, EOV1 and EOV2 tape labels shall exist after an EOT and a tape mark at the end of the tape. When a file ends within a tape, it shall be followed by a tape mark and then the file trailer labels EOF1 and EOF2.

5.2.1.2.2.2 Density and Blocking. Tapes shall be written at 6250 bits per inch (bpi) with the GCR recording method. The block length shall be denoted by the Block Length Field within a file's HDR2 label. Block size can vary from file to file. The allowed minimum tape block size shall be 2048 bytes while the maximum shall be 65534 bytes.

5.2.1.2.2.3 File Names. The allowable save set names in the VMS ANSI implementation used by SIF/DP shall be a subset of those in the original standard. Only the characters A through Z, 0 through 9, and the special characters '&', '-', '_', and '$' shall be used in save set names. The period may appear once within the name with a maximum of three characters following it. The save set name shall have no more than 17 characters.

5.2.2 SIF/DP File Formats. The file and data formats of the four main data sections shall be as detailed in the SSDB Data Base Design Document (DBDD). All files shall follow VAX/VMS conventions.

5.2.2.1 Header File

5.2.2.1.1 Header Data Encoding. The SIF/DP Data Base Header shall consist of one VAX/VMS save set, containing a single file. A compressed form of ASCII shall be used. The compression shall take the form of stripping all leading zeros and blanks from numeric strings and all leading and trailing blanks from character strings. Every ASCII field shall be a variable-length field, separated by the ASCII null character ('00'). Since fields are variable-length, records may also vary in length. Every record (except the file identifier record) shall begin with a 2-character keyword identifying its type.

5.2.2.1.1.1 Comment Records. The record keyword for a comment record shall be two consecutive asterisks (**). Following the keyword shall be the standard ASCII null character ('00') as the field separator. The comment field shall then continue until end of file (EOF) or the end of field separator ('00') is located in the SIF data file. Comment records shall not occur in the middle of any record in the file, but can be placed before or after any other record in the data file.

5.2.2.1.2 Header Section Structure. This section shall be the first save set on the first tape volume. The save set name of the SIF/DP Data Base Header shall be "SIFDP.HDR".

5.2.2.1.3 Header File Structure. The SIF/DP Data Base Header file format shall be as follows:

SIF File Identifier Record
Transmittal Description Record
Data Directory Record

2D Static Model Library Header File Name Record
for each 2D static model
    2D Static Model Entry Record

3D Static Model Library Header File Name Record
for each 3D static model
    3D Static Model Entry Record

3D Dynamic Model Library Header File Name Record
for each 3D dynamic model
    3D Dynamic Model Entry Record

for each culture cell
    Culture Cell Header Control Record
    for each manuscript within the cell
        Culture Manuscript Data File Names Record

for each terrain cell
    Terrain Cell Header Control Record
    for each manuscript within the cell
        Terrain Manuscript Data File Names Record

for each generic texture
    Generic Texture Entry Record

for each stage 3 specific model texture
    Stage 3 Specific Model Texture Entry Record

for each stage 2 specific model texture
    Stage 2 Specific Model Texture Entry Record

for each stage 1 specific model texture
    Stage 1 Specific Model Texture Entry Record

for each stage 3 specific areal texture
    Stage 3 Specific Areal Texture Entry Record

for each stage 2 specific areal texture
    Stage 2 Specific Areal Texture Entry Record

for each stage 1 specific areal texture
    Stage 1 Specific Areal Texture Entry Record

for each SMC/FDC texture
    SMC/FDC Texture Entry Record

5.2.2.1.3.1 SIF File Identifier Record. The field structure of this record shall be as follows:

File Identifier Field (always 'SIF/DP DATA BASE HEADER')

5.2.2.1.3.2 Transmittal Description Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'TD')
SIF Format Field
Originator Field
Recipient Field
Transmittal ID Field
Creation Date Field
Source Agency/Project Field
Database Name Field
Data On This Volume Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field
SIF Version Number Field

5.2.2.1.3.3 Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'DD')
Number of 2D Static Models Field
Number of 3D Static Models Field
Number of 3D Dynamic Models Field
Number of Culture Tiles Field
Number of Terrain Tiles Field
Number of Generic Textures Field
Number of Stage 3 Specific Model Textures Field
Number of Stage 2 Specific Model Textures Field
Number of Stage 1 Specific Model Textures Field
Number of Stage 3 Specific Areal Textures Field
Number of Stage 2 Specific Areal Textures Field
Number of Stage 1 Specific Areal Textures Field
Number of SMC/FDC Textures Field
Data Base SW Corner Field
Data Base NE Corner Field

5.2.2.1.3.4 2D Static Model Library Header File Name Record. This record shall be included when the SIF/DP data base includes 2D static models. The field structure of this record shall be as follows:

Record Keyword Field (always '2L')
File Name Field

5.2.2.1.3.5 2D Static Model Entry Record. The number of these records shall correspond to the number of 2D Static Models Field, found in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always '2S')
Model Data File Name Field
Model Number Field
Model Name Field
Model Description Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.2.2.1.3.6 3D Static Model Library Header File Name Record. This record shall be included when the SIF/DP data base includes 3D Static Models. The field structure of this record shall be as follows:

Record Keyword Field (always '3L')
File Name Field

5.2.2.1.3.7 3D Static Model Entry Record. The number of these records shall correspond tothe number of 3D Static Models Field, found in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always '3S')
Model Data File Name Field
Model Number Field
Model Name Field
Model Description Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.2.2.1.3.8 3D Dynamic Model Library Header File Name Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'DL')
File Name Field

5.2.2.1.3.9 3D Dynamic Model Entry Record. The field structure of this record shall be as follows:

Record Keyword Field (always '3D')
Model Data File Name Field
Model Number Field
Model Name Field
Model Description Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.2.2.1.3.10 Culture Cell Header Control Record. The number of these records shall correspond to the number of Culture Tiles Field in the Daa Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'CH')
Cell Header File Name Field
SW Corner Field
Number of Manuscripts Field

5.2.2.1.3.11 Culture Manuscript Data File Names Record. The number of these records shall correspond to the Number of Manuscripts Field in the Culture Cell Header Control Record. If a file does not exist, then the file name shall be represented by the null field. (The null field is indicated by consecutive null '00' separators.) The field structure of this record shall be as follows:

Record Keyword Field (always 'CM')
Manuscript File Name Field
Segment File Name Field
2-D Coordinate File Name Field
3-D Coordinate File Name Field
Error File Name Field
Model Reference Table File Name Field
Texture Reference File Name Field
SW Corner Field
NE Corner Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.2.2.1.3.12 Terrain Cell Header Control Record. The number of these records shall correspond to the Number of Terrain Tiles Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'TH')
Cell Header File Name Field
SW Corner Field
Number of Manuscripts Field

5.2.2.1.3.13 Terrain Manuscript Data File Names Record. The number of these records shall correspond to the Number of Manuscripts Field in the Terrain Cell Header Control Record. If a file does not exist, then the file name is represented by the null field. (The null field is indicated by consecutive null '00' separators.) The field structure of this record shall be as follows:

Record Keyword Field (always 'TM')
Manuscript File Name Field
Error File Name Field
SW Corner Field
NE Corner Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.2.2.1.3.14 Generic Texture Entry Record. The number of these records shall correspond to the Number of Generic Textures Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'GX')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Creation Date and Time Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.2.2.1.3.15 Stage 3 Specific Model Texture Entry Record. The number of these records shall correspond to the Number of Stage 3 Specific Model Textures Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'M3')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Capture Date and Time Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.2.2.1.3.16 Stage 2 Specific Model Texture Entry Record. The number of these records shall correspond to the Number of Stage 2 Specific Model Textures Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'M2')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Capture Date and Time Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.2.2.1.3.17 Stage 1 Specific Model Texture Entry Record. The number of these records shall correspond to the Number of Stage 1 Specific Model Textures Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'M1')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Capture Date and Time Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.2.2.1.3.18 Stage 3 Specific Areal Texture Entry Record. The number of these records shall correspond to the Number of Stage 3 Specific Areal Textures Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'A3')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Capture Date and Time Field
SW Corner Field
NE Corner Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.2.2.1.3.19 Stage 2 Specific Areal Texture Entry Record. The number of these records shall correspond to the Number of Stage 2 Specific Areal Textures Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'A2')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Capture Date and Time Field
SW Corner Field
NE Corner Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.2.2.1.3.20 Stage 1 Specific Areal Texture Entry Record. The number of these records shall correspond to the Number of Stage 1 Specific Areal Texture Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'A1')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Capture Date and Time Field
SW Corner Field
NE Corner Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.2.2.1.3.21 SMC/FDC Texture Entry Record. The number of these records shall correspond to the Number of SMC/FDC Textures Field in the Data Directory Record. The field structure of this record shall be as follows:

Record Keyword Field (always 'SF')
Image Sub-Header File Name Field
Image Data File Name Field
Texture Library Field
Texture ID Field
Texture Type Field
Horizontal Resolution Field
Vertical Resolution Field
Number of Texels Per Row Field
Number of Texels Per Column Field
Image Capture Date and Time Field
SW Corner Field
NE Corner Field
New Data Flag Field
Changed Data Flag Field
Security Classification Field
Control and Handling Field
Releasing Instructions Field
Classification Authority Field
Security Control Number Field
Security Downgrade Field
Downgrading Event Field

5.2.2.2 Model Data

5.2.2.2.1 Model Data Encoding. The SIF/DP model format is nearly identical with the formats of the SDBF Standard Simulator Data Base (SSDB). SIF/DP stores models in a dual format that includes both Constructive Solid Geometry (CSG) and polygonal geometry definitions. A SIF/DP model may have only the CSG definition, only the polygonal definition, or both. Other information, such as attributes, shall be stored only once for the model, regardless of the geometric definition(s) used. Together with the CSG definition of the basic model, the SSDB shall store instructions for automatically converting the model into various polygonal representations suitable for use on typical real-time image generators. The CSG commands stored in the SSDB, and used in SIF/DP shall comply with the CSG command language implemented by Interactive Computer Modelling Geometric Modelling System (ICMGMS). Polygons shall be stored with their corresponding attributes. Polygonal information may be included with or without a CSG definition of a model. Every record in the model data file shall begin with a 2-character keyword identifying its type.

5.2.2.2.2 Model Section Structure. Models in a SIF/DP data set shall be organized into three general classes, 2-D static models, 3-D static models, and 3-D dynamic models. Each class shall have a single library header file, which shall refer to separate Model Data Files containing the actual model representations. SIF/DP models shall be stored in up to nine levels of detail, numbered zero through eight. LOD 0 shall have the least amount of detail, while LOD 8 has the most detail. A series of tables shall be used to refer to colors, face-based texture references, vertex-to-vertex texture references, model-based texture references, user-defined FACS, and the SIF-defined FACS.

5.2.2.2.2.1 File Format. Each SIF model shall be described by a file made up of variable-length logical keyword records containing ASCII alphanumeric strings. This file shall consist of both geometry and attribute information, including all tables. The internal logical format of the string records shall vary in order to support a wide range of data about a model's geometry and attributes. If polygonal geometry exists, then polygon vertices shall exist.

5.2.2.2.2.2 File Content. When generating a SIF/DP data set from the SSDB, the inclusion of models may be toggled such that no models are included or all models are included. Each model library shall have its own header file. Every model shall be contained in its own file. Each model data file shall contain descriptions of all LOD (level of detail) versions of the model.

5.2.2.2.2.3 Save Sets. The SIF/DP Model Data Section shall consist of one or more save sets, as defined by the VAX/VMS Backup utility. The save set names shall be "MODELS_xxx", where "xxx" is the sequence number unique within the Model Data Section save sets.

5.2.2.2.3 Model File Structure. The file formats of all model files within a save set shall be in standard VAX/VMS format. The organization of each file shall be as described in the SSDB Data Base Design Document (DBDD) [PRC-2851-DBDD-3].

5.2.2.3 Culture Data. Spatially, features shall be represented as discrete points, lines, or surface polygons in two-dimensional space, defined in terms of geographic latitude and longitude. Each feature shall be described by mandatory, and possibly some optional, attributes. The SIF/DP shall store coordinates in resolution units of thousandths of an arc second.

5.2.2.3.1 Culture Data Encoding. A SIF/DP data set may include multiple LODs for any given area of coverage. These LODs shall correspond to feature resolutions of 100 meters, 30 meters, 10 meters, 3 meters, and 1 meter. Data based in SIF/DP shall include pointers between alternate representations of features at different LODs. Feature coding shall be based on the DMA FACS system.

5.2.2.3.2 Culture Section Structure. Culture data shall be maintained in tiled culture manuscripts, no larger than a one degree by one degree cell in horizontal extent. Data may be further physically subdivided into multiple levels of detail (LODs) for a given area, corresponding to the resolutions specified previously. SIF/DP files shall be managed along one degree cell boundaries. When a system receives or sends SIF/DP culture data, it shall receive or send all manuscripts and all LODs within a specified cell. The SIF/DP Culture Data Section shall consist of one or more save sets, as determined by the VAX/VMS Backup utility. The save set names shall be "CULTURE_xxx", where "xxx" is the sequence number, which is unique within the Culture Data Section save sets.

5.2.2.3.3 Culture File Structure. The file formats of all culture files within a save set shall be in standard VAX/VMS format. The organization of each file shall be as described in the SSDB Data Base Design Document (DBDD) [PRC-2851-DBDD-3].

5.2.2.4 Terrain Data

5.2.2.4.1 Terrain Data Encoding. A SIF data base may store multiple levels of detail (LODs) of terrain elevation data for any given area. Terrain resolutions shall be 3 arc seconds, 1 arc second, 0.3 arc seconds, and 0.03 arc seconds. Each elevation value shall be the height of the terrain above (or below) Mean Sea Level (MSL), expressed in resolution units of 0.1 meters.

5.2.2.4.2 Terrain Section Structure. Terrain elevation data shall be maintained in tiled manuscripts, which shall be no larger than a one degree by one degree cell in horizontal extent. Data may be further physically subdivided into multiple levels of detail (LODs) for a given area, corresponding to the resolutions specified previously. SIF/DP files shall be managed along one degree cell boundaries. When a system receives or sends SIF/DP terrain data, it shall receive or send all manuscripts and all LODs within a specified cell. The SIF/DP Terrain Data Section shall consist of one or more save sets, as determined by the VAX/VMS Backup utility. The save set names shall be "TERRAIN_xxx", where "xxx" is the sequence number unique within the Terrain Data Section save sets.

5.2.2.4.3 Terrain File Structure. The file formats of all terrain files within a save set shall be in standard VAX/VMS format. The organization of each file shall be as described in the SSDB Data Base Design Document (DBDD)[PRC-2851-DBDD-3].

5.2.2.5 Texture Data. If a SIF data base includes texture, it shall contain all of the producing system's texture within the specified area of coverage. If generic textures are included, then all generic textures in the sending system shall be included.

5.2.2.5.1 Texture Data Encoding. SIF/DP texture data shall be encoded as it is stored within the SSDB.

5.2.2.5.2 Texture Section Structure. A SIF/DP data base shall include between one and eight texture libraries defined as Stage 1 Areal, Stage 2 Areal, Stage 3 Areal, Stage 1 Model, Stage 2 Model, Stage 3 Model, Generic, and SMC/FDC. Attribute data shall be stored by logical groups; the actual texture data shall be stored in a gridded format. SIF/DP files shall be managed along one degree cell boundaries. When a system receives or sends SIF/DP texture data, it shall receive or send all manuscripts and all LODs within a specified cell. The SIF/DP Texture Data Section shall consist of one or more save sets, as determined by the VAX/VMS Backup utility. The save set names shall be "TEXTURE_xxx", where "xxx" is the sequence number, which is unique within the Texture Data Section save sets.

5.2.2.5.3 Texture File Structure. The file formats of all texture files within a save set shall be in standard VAX/VMS format. The organization of each file shall be as described in the SSDB Data Base Design Document (DBDD)[PRC-2851-DBDD-3].

5.2.3 SIF/DP Data Base Content. A SIF/DP data base shall include the full information content of the SSDB from which it is produced, limited only by the presence or absence of the various files identified in section 5.2.2 of this Standard.

6. NOTES

(This section contains information of a general or explanatory nature that may be helpful, but is not mandatory.)

6.1 Intended use. It is intended that this standard will be invoked by training simulator acquisition contracts in two ways: to specify the requirements for data bases to be delivered under those contracts, and to provide specifications for data bases to be furnished to the contractor as Government Furnished Property (GFP).

6.2 Acquisition requirements. Acquisition documents must specify the following:

a. Title, number, and date of the standard.

b. Issue of DoDISS to be cited in the solicitation, and if required, the specific issue of individual documents referenced (see section 2).

c. Whether or not alternative media is to be used.

6.3 Tailoring Guidance for Contractual Application. The following subparagraphs identify the points at which decisions can be made with regard to the selective application of this standard. The most basic decision, that is, the determination of whether this standard should be applied at all, needs to be continually reassessed during a program's lifecycle.

6.3.1 Prior to Contract. Sometimes, fundamental application decisions can be made even before a contract is awarded for a particular training simulator. These can often be included in the various activities leading up to the award of a contract, as discussed below.

6.3.1.1 RFP Preparation. Before going out with an RFP for a training simulator, the acquisition agency will usually have already identified the data base requirements of the system. It is at this time that the SDBF should become involved, through its knowledge of available SSDB holdings, and its ability to be applied to the training device being procured.

6.3.1.1.1 Selection of standard. The decision of which of two standards, SIF or GTDB, is most appropriate, will usually be highly dependent on the capabilities of the device being procured, as well as its development contractor. In general, a less complex device, or one being built by a contractor with few data base development tools, will use GTDB; whereas a high-performance system, or one being developed by a contractor with a strong data base development capability, will use SIF. The desire to obtain a SSDB-compatible copy of the data base is one reason to use SIF, but does not necessarily preclude the use of GTDB for the distribution of SDBF data sets to the contractor; in these cases, both standards may be used. Within the SIF standard, one also has the options of using the HDI format, the DP format, or even both; this decision will be driven by the computational system to be used for the DBGS, rather than any simulator-related characteristics.

6.3.1.1.2 Timing of Selection. In the case of some training system programs, data base requirements evolve during the course of the program, as part of the training requirements analysis process. Under such circumstances, this decision may need to be deferred until after contract award.

6.3.1.2 Submission of Proposal. Sometimes, it may be desirable to leave it to the individual bidders to propose which standard(s) will be applied, and if any SIF-formatted data bases will be delivered. This may be done when it is expected that some bidders will prefer one option over the other, or when it is likely that some will have greater data base generation capabilities than others. Giving the bidders this flexibility can, in some cases, result in decreased bid prices.

6.3.1.3 Source Selection. During the selection of a contractor, it may be possible to conduct a demonstration of the bidders' capabilities to produce and/or use SDBF-compatible data sets. Again, SDBF assistance should be obtained, as the provider of sample SIF and/or GTDB data sets for bidder consumption, as well as evaluator of bidder-produced SIF data sets. Bidders should be evaluated based upon their ability to produce and/or utilize these data sets in the most complete, efficient manner possible.

6.3.2 Contract Decisions. Unlike the application decisions identified above, certain alternatives can only be selected after the data base approach has evolved, as a part of the engineering process conducted under contract. The following subparagraphs identify some milestones at which SIF application decisions can be made.

6.3.2.1 Design Reviews. Some SIF alternatives can be formally selected when the contractor's data base system design is examined and approved at the preliminary design review (PDR) and/or critical design review (CDR).

6.3.2.1.1 Use of SDBF Sources. The use of SIF and/or GTDB as source material should be finalized by PDR. If it appears that the use of a SDBF-compatible source imposes a greater liability than its long-term benefit, there may be sufficient reason to waive the requirement, or change it in some way (for example, substituting GTDB for SIF, if the extraction of the needed information from the SIF data set proves too difficult for the contractor.) By PDR, the contractor will have gained enough understanding, through design analysis, to decide which approach is best for that program.

6.3.2.1.2 Population of Optional Fields. At contract award, the contractor is effectively given "carte blanche" to decide which, if any, of the many optional data items will be populated in any SIF data sets delivered. These should be formally reviewed and agreed upon by the contractor and Government, initially at PDR, and finally at CDR.

6.3.2.1.3 Exceptions to Standard. When SIF is invoked, exceptions to the mandatory provisions of the standard should not be granted, unless the contractor has positively demonstrated that full compliance will somehow result in an unsatisfactory product. The technical aspects of any such demonstration should be reviewed in the context of the overall system design, not later than CDR.

6.3.2.2 Data Base Working Groups. During the course of most simulator acquisition programs, a number of data base working group meetings are conducted, to review the identification, content, and design of training data bases. These meetings present convenient forums for low-level decisions, such as values to be assigned to User-Defined FACS codes for particular features. The utility of any required SIF output products should be continually reassessed; the data base working group meetings present handy points at which decisions can be made on whether or not to produce SIF data sets, and what they will contain, since the specific data base contents are formally reviewed at those times.

6.3.2.3 System Test. During the test phase of a trainer development, compliance with the SIF standard will be verified. Even at this late stage of the program, however, decisions relating to application of the SIF standard must be made. In general, these apply only to those contracts under which SIF data sets will be produced by the contractor.

6.3.2.3.1 Pre-Production Process Certification. Prior to committing to the full-scale production of SIF data sets, the contractor's DBGS should be certified as being SIF-compliant by the SDBF. By obtaining this certification, the producer can be exempted from the bulk of the time-consuming effort involved in the testing of individual data sets. This certification should be conducted well in advance of the planned start of the SIF conversion activity, so that the contractor has time to do rework, as necessary. (Note that the act of certification carries the implicit assumption that the SIF conversion will, in fact, be accomplished, verifying the capability to do so even before the corresponding data bases have been completed; in turn, this amplifies the importance of the decisions made at the data base working groups, as discussed above.)

6.3.2.3.2 Trainer Data Base Acceptance Testing. On contracts requiring delivery of SIF data sets, the final decision of whether or not to produce a specific data set should only be made after the content of the corresponding trainer data base has stabilized, and its quality has been proven. The acceptance testing of the individual data bases delivered with a simulator can serve as the ultimate decision points for the generation of the corresponding SIF data sets. These decisions should be made independently, i.e., on a data base-by-data base basis; it is entirely conceivable that a program may require the delivery of a number of data bases, some of which are suitable, and some which are not.

6.3.2.3.3 SIF Data Set Verification Testing. Although the certification of the DBGS precludes the need for much examination of individual SIF data sets, it will still be necessary to do some quality conformance inspections periodically. This may be done as a random sampling process, or focus specifically on certain high-interest data sets, depending on the requirements of the particular program.

6.4 Government-furnished property. When SIF is specified as a data base source for a specific acquisition, the contracting officer should arrange to furnish the contractor with a sample database for use in verification demonstrations and/or tests. The sample database should be obtained from the DoD Simulator Data Base Facility.

6.4.1 Testing Facilities. When SIF is specified as a contract deliverable, the contracting officer may wish to arrange access to a Government-controlled facility for use in verification demonstrations and/or tests. The contracting officer may also wish to arrange access to such a facility by the contractor to support developmental testing. These arrangements should also be made through the SDBF.

6.5 Data requirements. The following Data Item Descriptions (DIDs) must be listed, as applicable, on the Contract Data Requirements List (DD Form 1423) when this standard is applied on a contract, in order to obtain the data, except where DoD FAR Supplement 27.475-1 exempts the requirement for a DD Form 1423.

Reference Paragraph	DID Number	DID Title	Suggested Tailoring
4.4	DI-MCCR-80012	Software Design Document	Yes

The above DIDs were those cleared as of the date of this standard. The current issue of DoD 5010.12-L, Acquisition Management Systems and Data Requirements Control List (AMSDL), must be researched to ensure that only current, cleared DIDs are cited on the DD Form 1423.

6.6 Subject term (key word) listing.

Culture data
Constructive Solid Geometry (CSG)
Database standards
Feature data
Photo texture
Models
Project 2851
Simulator databases
Terrain data
Texture
Training systems

6.7 Referenced documents. The following documents were used as references, in preparation of this standard.

DEFENSE MAPPING AGENCY

DMA Standard Supporting Mark 90, Section 100, Glossary of Feature/Attribute Definitions, Second Edition, June 1988, revised December 1988.

(Application for copies should be adressed to Defense Mapping Agency, 8613 Lee Highway, Fairfax VA 22031-2137.)

DIGITAL EQUIPMENT CORPORATION

AA-LA06A-TE Guide to VMS Files and Devices, Appendix B, "VMS ANSI-Labeled Magnetic Tape," April 1988.

VMS Backup Utility Manual, April 1988

(Application for copies should be addressed to Digital Equipment Corporation, P.O. Box CS2008, Nashua NH 03061)

CUSTODIANS:
Army - PT
Navy - TD
Air Force - 11

PREPARING ACTIVITY:
Air Force - 11

(Project Nr. 69GP-0117)

APPENDIX A

This page intentionally left blank.

SIF DATA DICTIONARY

10. SCOPE

10.1 Scope. This Appendix is a mandatory part of the standard. The information contained herein is intended for compliance.

10.2 Purpose. This Appendix provides definitions of the data fields to be used within SIF data bases.

20. APPLICABLE DOCUMENTS

20.1 Government documents

20.1.1 Specifications, standards, and handbooks. The following specifications, standards, and handbooks form a part of this Appendix to the extent specified herein. Unless otherwise specified, the issues of these documents shall be those listed in the issue of the Department of Defense Index of Specifications and Standards (DoDISS) and supplement hereto, cited in the solicition (see 6.2 of this Standard).

MIL-STD-1820 Generic Transformed Data Base Design Standard

(Unless otherwise indicated, copies of federal and military specifications, standards, and handbooks are available from the Naval Publications and Forms Center, ATTN: NPODS, 5801 Tabor Avenue, Philadelphia PA 19120-5099.)

20.2 Order of precedence. In the event of a conflict between the text of this Appendix and the references cited herein, the text of this Appendix shall take precedence. Nothing in this Appendix, however, supersedes applicable laws and regulations unless a specific exemption has been obtained.

30 DEFINITIONS AND ACRONYMS

30.1 Definitions. The definitions provided in this Standard shall apply to this Appendix.

40 GENERAL REQUIREMENTS

40.1 This Appendix shall be a mandatory part of the standard. The information contained herein is intended for compliance.

40.2 Data Types. Data items shall be represented in the forms specified in Table A-1. Appendix C, para 60.1, explains Gridded Data Section (GDS) application and data item types of binary, integer, real, string, enumerated, and boolean.

40.3 GTDB Commonality. The definitions of some data items included within SIF data sets differ from the definitions contained within MIL-STD-1820, Generic Transformed Data Base Design Standard. Users of both SIF and GTDB need to exercise caution when using common software for both SIF and GTDB data sets.

50 DETAILED REQUIREMENTS

50.1 Data Items. All data items included in a SIF data set shall adhere to the definitions specified in Table A-2 of this standard.

60 NOTES

(This section contains information of a general or explanatory nature that may be helpful, but is not mandatory.)

6.1 Referenced documents. The following documents were used a references, in preparation of this Appendix.

AMERICAN NATIONAL STANDARDS INSTITUTE

ANSI X3.27	Information Systems - File Structure and Labeling of Magnetic Tapes for Information Interchange
ANSI/IEEE STD 754	Binary Floating Point Arithmetic

(Application for copies should be addressed to American National Standards Institute, 11 West 42nd Street, New York NY 10036.)

DEFENSE INTELLIGENCE AGENCY

DDM-2600-63220-90	National Imagery Transmission Format (NITF), Version 1.1, 1 March 1989, sections 1 through 4.5

(Application for copies should be addressed to Defense Intelligence Agency, DIA/DM-1A, 3100 Clarendon Boulevard, Arlington, VA 22201-5317.)

DEFENSE MAPPING AGENCY

DMA Standard Supporting Mark 90, Section 100, Glossary of Feature/Attribute Definitions, Second Edition, June 1988, revised December 1988.

(Application for copies should be adressed to Defense Mapping Agency, 8613 Lee Highway, Fairfax VA 22031-2137)

DIGITAL EQUIPMENT CORPORATION

AA-LA06A-TE Guide to VMS Files and Devices, Appendix B, "VMS ANSI-Labeled Magnetic Tape," April 1988.

VMS Backup Utility Manual, April 1988

(Application for copies should be addressed to Digital Equipment Corporation, P.O. Box CS2008, Nashua NH 03061)

U.S. DEPARTMENT OF COMMERCE

Initial Graphics Exchange Specification (IGES), Version 4.0, June 1988, sections applicable to CSG

(Application for copies should be adressed to U.S. Department of Commerce, National Bureau of Standards.)

INTERACTIVE COMPUTER MODELLING, INCORPORATED

General Information Manual, May 1988.

(Application for copies should be addressed to Interactive Computer Modelling, Inc, 12200 Sunrise Valley Drive, Suite 210, Reston VA 22091.)

PLANNING RESEARCH CORPORATION

PRC-2851-DBDD-3 Data Base Design Document (DBDD), Standard Simulator Data Base (SSDB), Project 2851 (F33657-86-C-0182)

PRC-2851-DBDD-5 Data Base Design Document (DBDD), Appendix I, Data Type Dictionaries for Project 2851 (F33657-86-C-0182)

(Application for copies should be addressed to PRC, 1500 Planning Research Drive, McLean VA 22102.)

(Non-Government standards and other publications are normally available from the organizations that prepare or distribute the documents. These documents also may be available in or through libraries or other informational services.)

Table A-1 Data Type Definitions

BINARY INT	Binary Integer Value (2's complement)	Atomic Value
BINARY REAL2D6	Binary Floating Point Value, Binary Floating Point Value (6 significant digits)	Composite Level
BINARY REAL3D6	Binary Floating Point Value, Binary Floating Point Value, Binary Floating Point Value (6 significant digits)	Composite Level
BOOLEAN	Enumerated Value	Atomic Level
ENUM	Enumerated Value	Atomic Level
HEX	Hexadecimal Value	Composite Level
INT	Integer Value	Atomic Level
INT2D	Integer Value, Integer Value	Composite Level
INT3D	Integer Value, Integer Value, Integer Value	Composite Level
INT4D	Integer Value, Integer Value, Integer Value, Integer Value	Composite Level
REAL6	Floating Point Value (6 significant digits)	Atomic Level
REAL10	Floating Point Value (10 significant dig.s)	Atomic Level
REAL2D6	Floating Point Value, Floating Point Value (6 significant digits)	Composite Level
REAL2D10	Floating Point Value, Floating Point Value (10 significant dig.s)	Composite Level
REAL3D6	Floating Point Value, Floating Point Value, Floating Point Value (6 significant digits)	Composite Level
REAL3D10	Floating Point Value, Floating Point Value, Floating Point Value (10 significant dig.s)	Composite Level
REAL4D6	Floating Point Value, Floating Point Value, Floating Point Value, Floating Point Value (6 significant digits)	Composite Level
STR	ASCII Text String values	Atomic Level

Field Name	Type	Length (CHARS)	Range	Description
2-D Geometric Correction Flag (Gridded Data Section (GDS))	BOOLEAN	5	TRUE, FALSE	Flag indicating whether an image has been positioned/corrected in 2-D space
2-D/3-D Coordinates Flag	ENUM	2	2D, 3D	Flag indicating the type of coordinates referenced by a culture segment
3-D Geometric Correction Flag (GDS)	BOOLEAN	5	T, F	Flag indicating whether an image has been positioned/corrected in 3-D space
Above Tile Neighbor ID (GDS)	INT	10	0..2147483647	The identifier of the neighboring model specific image above the current image; used only for Stage 3 model textures
Absolute Horizontal Accuracy	STR	16	--	Definition of horizontal accuracy standard applying to a data source product
Absolute Latitude/Longitude (GDS)	STR	22	HDDMMSSSSSb HDDDMMSSSSS	Actual ground location in absolute geodetic coordinates for areal texture footprint boundary point coordinates and for areal texture control points; H = hemisphere, DD or DDD = degrees, MM = minutes, SSSSS = thousandths of seconds, and b = blank character (" ")
Absolute Vertical Accuracy	STR	16	--	Definition of vertical accuracy standard applying to a data source product
Absorptivity	REAL6	12	0.0..1.0	Ratio of radiant (thermal) energy to the energy incident upon it
Accuracy	STR	16	--	Definition of accuracy standard applying to a data source product

Field Name	Type	Length (CHARS)	Range	Description
All Convex Clusters Flag	BOOLEAN	1	T, F	Indicates whether all clusters of a model are convex
All Convex Polygons Flag	BOOLEAN	1	T, F	Indicates whether all polygons of a model are convex
Articulated Part Flag	BOOLEAN	1	T, F	Indicates whether the referenced model is an articulated part
Attachment Level (GDS)	INT	3	0..998	Display level to which a new object is to be attached for editing purposes
Attribute Value	--	--	--	The value assigned to a FACS code. This value will differ based on the type of FACS code. The following list shows the values that can be assigned: one byte integer, two byte integer, four byte integer, real (six significant digits), long real (ten significant digits), integer pair (2 -four byte integers), integer triplet (3 - four byte integers), real pair (2 real numbers with six significant digits), real triple (3 real numbers with six significant digits), long real pair (2 long real numbers with ten significant digits), long real triple (3 long real numbers with ten significant digits), string, enumerated, boolean, or null value

Field Name	Type	Length (CHARS)	Range	Description
AV Code 1	STR	6	--	Additional Value Code 1; alphanumeric code assigned to models in conjunction with the Feature Descriptor Code to further classify models within a set of categories based on the DMA AV Codes as extended by P2851; AV Codes to be used shall be those that clearly identify the type of object being modeled (e.g., BFC053 categorizes a building as a court house; BFC007 categorizes a building as a house of worship and BWT004 further categorizes it as a church); AV Codes that simply describe an attribute of the object (e.g., roof pitch) should not be used
AV Code 2	STR	6	--	Additional Value Code 2; alphanumeric code assigned to models in conjunction with the Feature Descriptor Code to further classify models within a set of categories based on the DMA AV Codes as extended by P2851; refer to "AV Code 1" for further explanation
AV Code 3	STR	6	--	Additional Value Code 3; alphanumeric code assigned to models in conjunction with the Feature Descriptor Code to further classify models within a set of categories based on the DMA AV Codes as extended by P2851; refer to "AV Code 1" for further explanation
Base Polygon ID	INT	10	0..2147483647	ID number of polygon that can be used as base polygon; if non-existent, then it defaults to 0

Field Name	Type	Length (CHARS)	Range	Description
Beginning Coordinates	INT2D	23	(-2147483648.. 2147483647; -2147483648.. 2147483647)	Coordinates of the first point of a culture segment (latitude;longitude) in ten thousandths of arc seconds (relative to the southwest corner of the culture tile)
Below Tile Neighbor ID (GDS)	INT	10	0..2147483647	The identifier of the neighboring model specific image below the current image; used only for Stage 3 model textures
Boundary Coordinates (GDS)	STR	22	HDDMMSSSSSb HDDDMMSSSSS	Coordinates (in latitude and longitude) used to define the footprint of a texture image or a terrain file; H = hemisphere, DD and DDD = degrees, MM = minutes, SSSSS = thousandths of seconds, and b = blank (" ")
Boundary ID (BOTH)	INT	5	0..32767	A unique identifier for a terrain or texture footprint
Boundary Point ID (GDS)	INT	5	0..32767	A unique identifier for a point that is located on a terrain or texture footprint
Bounding Rectangle Coordinates	INT2D, INT2D	47	(-2147483648.. 2147483647; -2147483648.. 2147483647), (-2147483648.. 2147483647; -2147483648.. 2147483647)	The southwest and northeast corners of a minimum geodetic bounding rectangle around a superfeature, feature, or segment in ten-thousandths of seconds relative to the southwest corner of the tile
Calibrated Focal Length (GDS)	REAL10	16	10.0..10000.0	An adjusted value of the focal length computed to equalize the positive and negative values of lens distortion over the entire focal plane (expressed in millimeters)

Field Name	Type	Length (CHARS)	Range	Description
Calibrated Point of Symmetry Offset (GDS)	REAL10	16	-1.0..1.0	The adjusted position which gives the best symmetry of radial lens distortion. The indicated principal point is formed by the intersection of lines connecting the fiducial marks. The calibrated point of symmetry offset is the difference between these two points (expressed in meters)
Calibrated Principal Point Offset (GDS)	REAL10	16	-1.0..1.0	The foot of the perpendicular from the interior perspective center to the focal plane. The indicated principal point is formed by the intersection of lines connecting the fiducial marks. The calibrated principal point offset is the difference between these points (expressed in meters)
Calibration Report Image Coordinates (GDS)	REAL2D10	33	(-1.0..1.0, -1.0..1.0)	The position of the fiducial marks as determined by a laboratory calibration of the camera (expressed in meters)
Camera Position in Height (GDS)	REAL10	16	0.0.. 1.393796575e+42	Height of the camera above mean sea level expressed in meters
Camera Position in Lat/Lon (GDS)	STR	22	HDDMMSSSSSb HDDDMMSSSSS	Geographic location of the sensor used to capture the data expressed in thousandths of arc seconds in absolute coordinates; H = hemisphere, DD or DDD = degrees, MM = minutes, SSSSS = thousandths of seconds, and b = blank character (" ")
Cell Boundary	STR	14	HDDHDDDHDDHDDD	Southwest and northeast corners of a geodetic bounding box that contains all of the culture data for a tile, expressed in absolute ground coordinates

Field Name	Type	Length (CHARS)	Range	Description
Changed Data Flag	BOOLEAN	1	T, F	Flag indicating whether a model, culture tile, terrain tile, or texture originally from the SSDB has been changed (ignored if New Data Flag is T)
Classification Authority	STR	20	--	The identity of the classification authority for a SIF database or elements within a SIF database. The code shall be in accordance with the regulations governing the appropriate security channels
Clipped Boundary Flag	BOOLEAN	1	T, F	Flag indicating whether a culture feature has been clipped to the edge of a tile
Cluster ID	HEX	32	0..F (32 times)	Unique identifier assigned to clusters of a model. It is a 32 digit hexadecimal number.
Codewords	STR	40	--	Security compartments associated with an image. Digraphs in accordance with DIAM 65-19 and its supplements, trigraphs not contained in DIAM 65-19, and complete codewords or project numbers may be used. The selection of a relevant set of codewords is implementation specific. The individual values are separated by single spaces
Collection System (BOTH)	STR	10	--	ID of the system used to collect data

Field Name	Type	Length (CHARS)	Range	Description
Color Definition Type	ENUM	3	HCV, RGB	Color definition system used
Color Description	STR	10	--	Description of a color
Color Table File Name	STR	17	--	The name of the color table file included with SIF/HDI models or SIF/HDI culture data
Color Table Index	INT	10	0..2147483647	Index of a color in the color table
Compilation Criteria (BOTH)	STR	160	--	Description of data capture criteria
Compilation Date (BOTH)	STR	6	YYMMDD	Date of data compilation; YY = year, MM = month, DD = day
Component ID	INT	4	0..1000	Unique ID number assigned to a component of a model
Compression Rate Code (GDS)	STR	4	--	A code indicating the compression rate for the image. If Image Compression = C0, the code is user defined. If Image Compression = C1, the codes are as follows: 1D = 1 Dimensional Coding; 2DS = 2 Dimensional Coding Standard Vertical Resolution, K=2; 2DH = 2 Dimensional Coding High Vertical Resolution, K=4. If Image Compression = C2, the Compression Rate Code is given in the form n.nn, representing the average number of bits per pixel over the image after compression. Valid codes are 0.75, 1.40, 2.30, and 4.50. Omitted if Image Compression is NC.

Field Name	Type	Length (CHARS)	Range	Description
Control and Handling	STR	40	--	Security handling instructions associated with a SIF tape, or elements within the SIF/HDI database
Control Point ID (GDS)	INT	5	0..32767	Numeric identifier of a specific control point
Control Point Name (GDS)	STR	40	--	Field used to associate a textual name with a control point
Convex Cluster Flag	BOOLEAN	1	T, F	Indicates whether a cluster in a model is convex
Convex Polygon Flag	BOOLEAN	1	T, F	Indicates whether a polygon in a model is convex
Coordinate Field	BINARY REAL2D6	8	(-1.9342 e+25.. 1.9342 e+25; -1.9342 e+25.. 1.9342 e+25)	For model data, this value is used to define the x and y location in meters of a vertex; value is 2D for 2D static models; represented in standard ANSI/IEEE binary floating point notation (ANSI/IEEE Std 754-1985)
	BINARY REAL3D6	12	(-1.9342 e+25.. 1.9342 e+25; -1.9342 e+25.. 1.9342 e+25; -1.9342 e+25.. 1.9342 e+25)	For model data, this value is used to define the x, y and z location in meters of a vertex or the values of a vertex normal for a given vertex; value is 3D for 3D static and 3D dynamic models; represented in standard ANSI/IEEE binary floating point notation (ANSI/IEEE Std 754-1985)
Correlation Priority	INT	3	0..127	A number indicating the relative importance of a feature or vertex for maintaining correlation among components in a culture tile. Higher numbers indicate greater priority

Field Name	Type	Length (CHARS)	Range	Description
Counter-Clockwise Areals Flag	BOOLEAN	1	T, F	Flag indicating whether all areal features in the culture database are counter-clockwise
Creation Date	STR	6	YYMMDD	The date that a SIF tape is created, where YY = year, MM = month, and DD = day
Culture Centroid	INT2D	23	(-2147483648.. 2147483647; -2147483648.. 2147483647)	The center of a feature object where the latitude and longitude are expressed in ten thousandths of an arc second
	INT3D	35	(-2147483648.. 2147483647; -2147483648.. 2147483647; -2147483648.. 2147483647)	The center of a feature object where the latitude and longitude are expressed in ten thousandths of an arc second and the elevation in tenths of a meter
Culture Coordinate System	ENUM	14	GEODETIC, GEOCENTRIC, MERCATOR, TRANS_MERC, LAMBERT, POLAR, LOCAL, GEODETIC_FLOAT	The name of the coordinate system with which the data are defined

Field Name	Type	Length (CHARS)	Range	Description
Cycle Rate Off Time	INT	10	0..2147483647	The period of time which a light remains in the off state, expressed in thousandths of seconds
Cycle Rate On Time	INT	10	0..2147483647	The period of time which a light remains in the on state, expressed in thousandths of seconds
Data Base Header File Name	STR	17	--	The name of the database header file included with SIF culture data
Data Base NE Corner	STR	24	HDDMMSSSSSb HDDDMMSSSSSS	The northeast corner of the coverage contained within a SIF database, where HDDMMSSSSSS = hemisphere (N,S), degrees, minutes and ten thousandths of seconds of latitude; b = blank (" "); and HDDDMMSSSSSS = hemisphere (E,W), degrees, minutes and ten thousandths of seconds of longitude
Data Base Sentinel (GDS)	ENUM	7	SIF/HDI, SIF/DP, GTDB	Label identifying the data base; used by the NITF data files

Field Name	Type	Length (CHARS)	Range	Description
Data Base SW Corner	STR	24	HDDMMSSSSSSb HDDDMMSSSSSS	The southwest corner of the coverage contained within a SIF database, where HDDMMSSSSSS = hemisphere (N,S), degrees, minutes and ten thousandths of seconds of latitude; b = blank (" "); and HDDDMMSSSSSS = hemisphere (E,W), degrees, minutes and ten thousandths of seconds of longitude
Data Edition Number	STR	3	--	The edition number of the database
Data On This Volume Flag	BOOLEAN	1	T, F	Flag indicating whether the database files for a SIF database start on the same tape as the SIF Data Base Header File (which is always the first file on the first volume)
Data Range	--	--	--	Defines the valid range for a data type that is used to define User-Defined FACS Code. The definition of the data range depends on the data type. For I1, I2, I4, R6, R10, I2D, I3D, R2D6, R3D6, R2D10, R3D10 data types, their data ranges are defined by a minimum value and a maximum value. For STR data type, its data range is defined by the length of the string. For ENU data type, its data range is defined by a list of valid symbols that constitutes the enumerated type. FLA and NO types do not require data ranges.
Data Series Designator	STR	5	--	Product Identifier
Data Source Table File Name	STR	17	--	The name of the data source table file included with SIF/HDI models
Data Source Table Index	INT	3	1..127	Index of a data source in a data source table

Field Name	Type	Length (CHARS)	Range	Description
Database Name	STR	80	--	Textual name associated with a SIF culture database
Default Source Identifier	INT	5	1..32767	Index of a data source in a data source table
Diffuse Reflectance	REAL6	12	0.0..1.0	Radar backscatter coefficient, expressed as a ratio
Directionality	REAL10	16	0.0..360.0	Angle from north by which a point light is visible. A value of '360.0' indicates that the light is omnidirectional
Directivity	ENUM	4	UNI, BI, OMNI	Indicator of shape of the planar response curve of a feature or model to a sensor (visual response)
Directivity (Infrared)	ENUM	4	UNI, BI, OMNI	Indicator of shape of the planar response curve of a feature or model to a sensor (infrared response)
Directivity (Radar)	ENUM	4	UNI, BI, OMNI	Indicator of shape of the planar response curve of a feature or model to a sensor (radar response)
Display Level (GDS)	INT	3	0..999	Unique graphic display level of an image relative to other message components in a composite. A higher number means that the item is to be displayed in front of other items with lower display level values
Downgrading Event	STR	40	--	The description of the event which caused a SIF database or elements within a SIF database to be downgraded

Field Name	Type	Length (CHARS)	Range	Description
East Tile Neighbor ID (GDS)	INT	10	0..2147483647	The identifier of the neighboring areal specific image to the east; used only for Stage 3 areal textures
Elevation Accuracy Standards (GDS)	STR	80	--	Description of the accuracy standards used to capture the terrain information
Elevation	BINARY INT	4	-12000000.. 100000	The z value that is used to express elevation for a 3-D coordinate in the culture 3-D vertex file, expressed in millimeters
Elevation Resolution (GDS)	ENUM	11	METERS, CENTIMETERS	Flag indicating whether 16 bit or 24 bit terrain post values are being provided in the SIF/HDI terrain file
Elevation Value (GDS)	BINARY INT	2	-12000..10000	Elevation values to be used when the Elevation Resolution is identified as being meters
		3	-1200000.. 1000000	Elevation values to be used when the Elevation Resolution is identified as being centimeters
Emissivity	REAL6	12	0.0..1.0	Ratio of the rate of IR radiation from a feature or model as a consequence of its temperature only, to the corresponding rate of emission from a blackbody at the same temperature
Encryption (GDS)	INT	1	0..1	Flag that indicates whether an image is encrypted. 0 = no encryption, 1 = encrypted
Ending Coordinates	INT2D	23	(-2147483648.. 2147483647; -2147483648.. 2147483647)	Coordinates of the last point of a culture segment (latitude;longitude) in ten thousandths of arc seconds (relative to the southwest corner of the culture tile)

Field Name	Type	Length (CHARS)	Range	Description
Enumerated Item Name	STR	80	--	ASCII string representing a valid value for a new enumerated type for a user-defined FACS attribute
Exitance	REAL6	12	(0.0.. 1.93428 E+25)	Rate of flow of infrared radiation from a feature or model polygon per unit of surface area expressed in watts/cm**2
Explicit Closure of Areals Flag	BOOLEAN	1	T, F	Flag indicating whether all areal features are explicitly closed or implicitly closed in the culture database
Extended Header Data Length (GDS)	INT	5	0..99999	The length in bytes of the SIF/HDI User Defined Header Data Record
Extended Sub-Header Data Length (GDS)	INT	5	0..99999	The length in bytes of the SIF/HDI User Defined Image Data or the SIF/HDI User Defined Terrain Data
Face Based Texture Reference Table File Name	STR	17	--	The name of the face based texture reference table file included with SIF/HDI models
FACS Attribute Code	STR	6	See DMA FACS Glossary and SIF Specific FDCs	Attribute codes that can be used for additional descriptors for culture features or models

Field Name	Type	Length (CHARS)	Range	Description
FACS Class	ENUM	5	I1, I2, I4, I2D, I3D, R6, R10, R2D6, R2D10, R3D6, R3D10, FLA, STR, ENU, NO	Specifies the data type based on which a user-defined FACS code is defined. I1 represents integer type of one byte. I2 represents integer type of two bytes. I4 represents integer type of four bytes. R6 represents real type with six significant digits. R10 represents real type with ten significant digits. I2D represents a 2-D integer pair where the elements are of I4 type. I3D represents a 3-D integer triplet where the elements are of I4 type. R2D6 represents a 2-D real pair where the elements are of R6 type. R2D10 represents a 2-D real pair where the elements are of R10 type. R3D6 represents a 3-D real triplet where the elements are of R6 type. R3D10 represents a 3-D real triplet where the elements are of R10 type. STR represents string type. ENU represents enumerated type, FLA represents boolean type. NO indicates that the user-defined FACS has no associated value.
FACS Description	STR	160	--	Textual description of a User-defined FACS code
FACS Table File Name	STR	17	--	The name of the FACS table file included with SIF/HDI models or with SIF/HDI culture data
FACS Table Index	INT	5	0..32767	Index of a FACS entry in a FACS table
Feature Description	STR	10	--	Description of a feature

Field Name	Type	Length (CHARS)	Range	Description
Feature Descriptor Code	STR	5	--	Alphanumeric code assigned to classify culture and models within a set of hierarchical categories, based on the DMA FACS as extended by P2851
Feature File Name	STR	17	--	The name of the feature file included with SIF/HDI culture data
Feature Fragment Flag	BOOLEAN	1	T, F	Flag indicating whether the feature in question has been fragmented along (clipped to) the tile boundary
Feature Identification Code	ENUM	4	NO, F101, F102, F103, F104, F105, F106, F110, F111, F112, F113, F114, F115, F116, F120, F121, F122, F130, F135, F136, F137, F138, F145, F146, F150, F151, F152, F153, F154, F155, F156, F160, F161, F162, F163, F164, F165, F166, F170, F171, F172, F173, F174, F180, F181, F182, F183, F184, F185, F186, F187, F188, F189, F190,	Unique feature identifier within the culture tile. Values for this may be derived using the DMA FID codes identified in the Digital Landmass System (DLMS) DFAD specification, or by the SIF implementor. If the SIF implementor supplies non-standard (non-DFAD) FID codes, the FID/FDC cross reference table must be supplied

Field Name	Type	Length (CHARS)	Range	Description
Feature Identification Code (continued)	ENUM	4	F201, F202, F203, F204, F205, F206, F207, F208, F209, F220, F221, F222, F223, F224, F230, F231, F232, F233, F234, F235, F236, F237, F238, F239, F240, F244, F245, F250, F251, F252, F253, F254, F255, F260, F261, F262, F263, F264, F265, F267, F270, F271, F272, F273, F274, F275, F276, F277, F280, F281, F282, F283, F290, F301, F302, F303, F304, F305, F320, F321, F322, F323, F324, F325, F330, F331, F332, F334, F340, F341, F343, F344, F350, F352, F401, F402, F403, F420, F421, F430, F433, F434, F435, F436, F450, F451, F501, F511, F512, F520, F521, F530, F531, F532, F535, F536, F540, F541, F542, F543, F544, F560, F561, F601, F602, F603, F604, F605, F606, F610, F620,	

Field Name	Type	Length (CHARS)	Range	Description
Feature Identification Code (continued)	ENUM	4	F621, F622, F630, F631, F632, F640, F641, F650, F680, F681, F682, F683, F684, F701, F702, F703, F704, F705, F706, F707, F710, F711, F712, F713, F714, F715, F716, F717, F718, F719, F720, F721, F725, F750, F751, F861, F862, F863, F864, F865, F901, F902, F906, F907, F908, F909, F910, F911, F912, F913, F914, F915, F916, F920, F921, F922, F923, F924, F925, F926, F927, F928, F929, F930, F931, F932, F933, F934, F940, F941, F942, F943, F944, F945, F946, F947, F948, F949, F950, F951, F952, F953, F954, F955, F956, F960, F961, F962, F963, F964, F965, F966, F967, F980	
Feature Number	INT	10	1..2147483647	Unique feature identifier within a culture tile
Feature Onset	BOOLEAN	1	T, F	Indicator for changing radar backscatter coefficients

Field Name	Type	Length (CHARS)	Range	Description
FID/FDC Reference Table File Name	STR	17	--	The name of the FID/FDC reference table file included with SIF/HDI culture data
File Identifier	STR	80	--	Alphanumeric string identifying the file type of a SIF file; always in the first record of every ASCII file
File Name Field	STR	17	--	The name of a data file which is located on the SIF tape.
Film Quality (GDS)	STR	20	--	Quality of film used to capture the image
Fixed Order Priority	INT	10	0..2147483647	Number used in determining the order of display of polygons within a model; used for hidden surface computations
Generic Model Flag	BOOLEAN	1	T, F	Indicates whether the model is generic
Generic Texture Set Name (GDS)	STR	20	--	Textual identifier identifying a set of generic textures that represent the same entity, where each member of the set has a different size and/or resolution
Geographic Location Name (GDS)	STR	40	--	A textual name associated with an areal specific image or SMC/FDC image
Geographic Tie Point ID (GDS)	INT	10	0..2147483647	A unique identifier of a geographic tie point

Field Name	Type	Length (CHARS)	Range	Description
Global Reference Point	INT2D	23	(-2147483648.. 2147483647; -2147483648.. 2147483647)	A point on 2D culture which corresponds to the origin of the texture being mapped
	INT3D	35	(-2147483648.. 2147483647; -2147483648.. 2147483647; -2147483648.. 2147483647)	A point on 3D culture or 3D polygonized terrain which corresponds to the origin of the texture being mapped
Haze Removal Flag (GDS)	BOOLEAN	5	T, F	Flag indicating whether haze has been removed from an image
Highest Feature Number	INT	10	1..2147483647	Identifier of the highest feature number contained within a culture tile
Highest Segment Number	INT	10	1..2147483647	Identifier of the highest segment number contained within a culture tile
Horizontal Captured Texel Size (GDS)	REAL10	16	0.0.. 1.393796575e+42	Approximate ground distance for a texel (expressed in meters) in the horizontal x-direction
Horizontal Resolution (BOTH)	REAL6	12	0.0..1.93428e+25	Horizontal length c' a texel; units are arc-seconds/texel for Stage 3 Areal Texture and meters/texel for all others.
Horizontal Size (GDS)	REAL6	12	0.0..1.93428e+25	The horizontal size of the entire image in meters, e.g., 1000.0 Meters
IGES Sequence Number for Component	INT	4	0..2000	Identifier for an IGES record that defines a component
Image Capture Date and Time (BOTH)	STR	12	YYMMDDHHMMSS	The date and time of day that a SIF image was captured, where YYMMDD = Year, Month and Day, and HHMMSS = Hours (0..24), Minutes and Seconds

Field Name	Type	Length (CHARS)	Range	Description
Image Classification Authority (GDS)	STR	20	--	The identity of the classification authority for an image. The code shall be in accordance with the regulations governing the appropriate security channels
Image Codewords (GDS)	STR	40	--	Security compartments associated with an image. Digraphs in accordance with DIAM 65-19 and its supplements, trigraphs not contained in DIAM 65-19, and complete codewords or project numbers may be used. The selection of a relevant set of codewords is implementation specific. The individual values are separated by single spaces
Image Comment (GDS)	STR	80	--	Field to be used for free form comments. May be used for image specific information. If the comment is classified, then it will be preceded by the classification, including codeword(s). Omitted if Number of Image Comments is zero.
Image Compression (GDS)	ENUM	2	NC, C0, C1, C2	If the image is transmitted in a compressed form, the letter C followed by a number between 0 and 2 is used to indicate the compression scheme used (C0 = compressed with a user specified algorithm, C1 = one bit, C2 = ARIDPCM). Given as NC if the image is not compressed
Image Control and Handling (GDS)	STR	40	--	Security handling instructions associated with an image

Field Name	Type	Length (CHARS)	Range	Description
Image Coordinate System (GDS)	ENUM	1	G, O	Coordinate system of the image where G = geodetic, O = Other. While NITF allows other values, P2851 has restricted the range of this field; for texture to be accepted into the active SSDB, the coordinate system must be geodetic
Image Coordinates (GDS)	INT2D	11	(0..99999, 0..99999)	X and Y location within an image
Image Data File Name	STR	17	--	The name of an image data file included with SIF texture data
Image Date & Time (GDS)	STR	14	DDHHMMSSZMONYY	Time (Zulu) of acquisition of the image where DD is the day of the month, HH is the hour, MM is the minute, SS is the second, the character Z, MON is the first three characters of the month, and YY is the year
Image Downgrading Event (GDS)	STR	40	--	If the Image Security Downgrade equals "999998" then this field must be present and must specify the event
Image File Creation Date and Time (GDS)	STR	12	YYMMDDHHMMSS	The date and time of day that a SIF image was created, where YYMMDD = Year, Month and Day, and HHMMSS = Hours (0..24), Minutes and Seconds
Image Filter Condition (GDS)	ENUM	1	N	Other values are reserved for future use

Field Name	Type	Length (CHARS)	Range	Description
Image Geographic Location (GDS)	STR	91	DDMMSSSSSXb DDDMMSSSSSY (4 times)	Geographic location of the image in geodetic coordinates. Geodetic coordinates are given as the latitude and longitude of the four corners in clockwise order beginning with the top left corner of the image as it is transmitted, where DDMMSSSSSX represents degrees, minutes, and thousandths of seconds of latitude with X= N or S for north or south, and DDDMMSSSSSY represents degrees, minutes, and thousandths of seconds of longitude with Y = E or W for east or west. b = blank (" "). Omitted if Image Coordinate System equals Other. P2851 has altered the size and accuracy to meet simulation requirements
Image ID (GDS)	STR	10	--	Textual identification of the image; unique across the entire SIF data base
Image Location (GDS)	STR	10	RRRRRCCCCC	An ordered pair defining the location in cartesian coordinates where the first pixel of the first line of the image is to be located, where RRRRR is the row and CCCCC is the column where the upper left corner of the image is to be located. (Not used by SIF)
Image Magnification (GDS)	STR	4	--	The magnification (or reduction) factor of the transmitted image relative to the original source image
Image Mode (GDS)	ENUM	1	S, I	Flag indicating band sequential ("S") or band interleaved ("I") transmission format. The SIF default is "S"

Field Name	Type	Length (CHARS)	Range	Description
Image Quality Comment (GDS)	STR	80	--	Free text comment field to include information pertaining to the quality of the image
Image Quality Rating (GDS)	ENUM	9	EXCELLENT, GOOD, FAIR, POOR	Rating of the quality of the image based on clarity and content of the image
Image Releasing Instructions (GDS)	STR	40	--	A list of countries and/or groups of countries to which the data are authorized for release.
Image Security Classification (GDS)	ENUM	1	T, S, C, R, U	Classification of the image and image sub-header. T = Top Secret, S = Secret, C = Confidential, R = Restricted, U = Unclassified
Image Security Control Number (GDS)	STR	20	--	Security control numbers associated with the image. The format is in accordance with the regulations governing the appropriate security channel(s)
Image Security Downgrade (GDS)	STR	6	--	An indicator which designates the point in time at which a declassification or downgrading action is to take place.
Image Source (GDS)	STR	80	--	Description of the source of the image. If the source is classified, then it will be preceded by the classification, including codeword(s)
Image Sub-Header File Name	STR	17	--	The name of an image sub-header file included with SIF texture data

Field Name	Type	Length (CHARS)	Range	Description
Image Sync Code (GDS)	INT	1	0, 4	A field that indicates whether a synchronization code has been provided for uncompressed or ARIDPCM compressed data
Image Title (GDS)	STR	80	--	Title of the image
Image Type (GDS)	STR	8	--	The type of image, such as BW for black and white, TV, SAR, XRAY, MS for multispectral, FAX for facsimile, or IR. Multispectral may be further denoted by TM7 for Thematic Mapper band 7.
Image-to-Image Contrast Enhancement Flag (GDS)	BOOLEAN	5	T, F	Flag indicating whether image to image contrast enhancements have been performed
Inner Image Contrast Enhancement Flag (GDS)	BOOLEAN	5	T, F	Flag indicating whether contrast enhancements have been performed within this image
Internal Material Category	INT	5	1..32767	Category code for material internal to an object
Internal Material Volume	REAL6	12	0.0..1.93428 e+25	Amount of material internal to an object, in liters
Island Number	INT	5	1..32767	Unique identifier within a culture tile of an area of common data resolution
Kappa (GDS)	REAL6	12	0.0..360.0	A rotation angle around the z-axis. A positive angle rotates the x-axis toward the y-axis (expressed in degrees)
Last Maintenance Date	STR	6	YYMMDD	The date when a model was last maintained, where YY = year, MM = month, and DD = day

Field Name	Type	Length (CHARS)	Range	Description
Last Maintenance Date and Time (GDS)	STR	12	YYMMDDHHMMSS	The last date and time of day that a SIF image was modified, where YYMMDD = Year, Month and Day, and HHMMSS = Hours (0..24), Minutes and Seconds
Latitude/Longitude	STR	22	HDDMMSSSSSb HDDDMMSSSSS	Ground location in absolute coordinates that is being used for an image control point, where H = hemisphere, DD or DDD = degrees, MM = minutes, SSSSS = thousandths of seconds, and b = blank (" ")
Latitude/Longitude	INT2D	23	(-2147483648.. 2147483647; -2147483648.. 2147483647)	Ground coordinates that are to be used for the identification of: a point on the boundary of the culture database, a point on the boundary of a data resolution island, or a point on the boundary of a culture tile in the database
Layer Number	INT	10	0..2147483647	A relative priority number indicating the sequence in which overlapping culture features, overlapping model polygons, or overlapping textures should be rendered for simulation. Higher values indicate a higher display priority (visual)
Layer Number (Infrared)	INT	10	0..2147483647	A relative priority number indicating the sequence in which overlapping culture features, overlapping model polygons, or overlapping textures should be rendered for simulation. Higher values indicate a higher display priority (infrared)

Field Name	Type	Length (CHARS)	Range	Description
Layer Number (Radar)	INT	10	0..2147483647	A relative priority number indicating the sequence in which overlapping culture features, overlapping model polygons, or overlapping textures should be rendered for simulation. Higher values indicate a higher display priority (radar)
Left Tile Neighbor ID (GDS)	INT	10	0..2147483647	The identifier of the neighboring model specific image to the left of the current image; used only for Stage 3 model textures
Length	REAL10	16	0.0..1.9342 e+25	The long dimension of a point or point light object in meters
Length of Image (GDS)	INT	10	0..1073741824	The length in bytes of the image. If the image is compressed, the length after compression is given
Length of Image Sub-Header (GDS)	INT	6	0..111000	The length in bytes of the image sub-header
Length of Terrain File (GDS)	INT	10	0..1073741824	The length in bytes of the terrain file
Length of Terrain Sub-Header (GDS)	INT	6	0..111000	The length in bytes of the terrain sub-header
Light Horizontal Center	REAL10	16	0.0..360.0	Angular offset to the center of light lobe, in degrees
Light Horizontal Fall	REAL10	16	0.0..360.0	Angle in which light intensity falls off, in degrees
Light Horizontal Width	REAL10	16	0.0..180.0	Half angle of lobe width from center, in degrees
Light Intensity	INT	5	0..32767	Candlepower of light, in candles

Field Name	Type	Length (CHARS)	Range	Description
Light String Shape	ENUM	3	LIN, CUR	Shape of a point light string (LINEAR or CURVILINEAR)
Light Type	ENUM	4	RW, APPR, VASI, TAXI, BEAC, STRO, OBST, CULT, AIRC, OTH, NO	Type of a light (RUNWAY, APPROACH, VASI, TAXI, BEACON, STROBES, OBSTRUCTION, CULTURAL, AIRCRAFT, OTHER, NONE)
Light Vertical Center	REAL10	16	0.0..360.0	Angular offset to the center of light lobe, in degrees
Light Vertical Fall	REAL10	16	0.0..360.0	Angle in which light intensity falls off, in degrees
Light Vertical Width	REAL10	16	0.0..180.0	Half angle of lobe width from center, in degrees
Linear Feature Texture Orientation	ENUM	4	PARA, PERP, RAND, OTH	Orientation of a texture in relationship to a linear feature (e.g., a flowing water texture would be parallel to the direction of a river feature). PARA means the texture y-axis is aligned with the direction of the linear feature, while PERP means the texture x-axis is aligned with the direction of the linear feature. RAND means randomly oriented.
LL Corner X/Y Image Coordinates (GDS)	INT2D	15	(-999999.. 999999, -999999.. 999999	X/Y cartesian coordinates of the lower left corner of the image
LOD Resolution Description	STR	80	--	Textual description of a model LOD resolution (e.g., in meters, number of polygons, etc.)

Field Name	Type	Length (CHARS)	Range	Description
Long Lineal	BOOLEAN	1	T, F	Reference to a point feature which could potentially look like a long linear feature by radar
Low Level Effects	BOOLEAN	1	T, F	Indicates normalcy to a terrain plate and therefore is an indication of higher radar backscatter
LR Corner X/Y Image Coordinates (GDS)	INT2D	15	(-999999.. 999999, -999999.. 999999	X/Y cartesian coordinates of the lower right corner of the image
LUT Entry Data (GDS)	STR	7	--	Value within a look up table (LUT) used for SMC/FDC texture only (SIF extends the NITF limit of 1 byte to 7 bytes and uses ASCII alphanumeric characters rather than binary data)
Maintenance Date	STR	6	YYMMDD	The date when a culture tile was last maintained, where YY = year, MM = month, and DD = day
Manuscript Boundary	INT2D, INT2D	43	(-324000000.. 324000000; -648000000.. 648000000), (-324000000.. 324000000; -648000000.. 648000000)	Two latitude/longitude coordinate pairs defining the southwest and northeast corners of the minimum bounding geodetic rectangle for the manuscript, in thousandths of seconds and absolute coordinates
Manuscript ID	STR	25	--	Unique textual name for a culture tile
Maximum Edges Per Polygon	INT	3	0..150	The maximum number of edges allowed in a polygon. 0 represents no limit
Maximum Height	REAL6	12	-1.93428e+25.. 1.93428e+25	Maximum height of a model expressed in meters

Field Name	Type	Length (CHARS)	Range	Description
Mean Square Error Latitude/Longitude/Height (GDS)	STR, STR, REAL10	38	DDDMMSSSSS, DDDMMSSSSS, 0.0.. 1.393796575e+42	A measure of accuracy which includes both bias squared and variance ; DDD = degrees, MM = minutes, SSSSS = thousandths of seconds, and the REAL10 value is in meters squared
Mean Square Error Omega/Phi/Kappa (GDS)	STR, STR, REAL10, STR	49	DDDMMSSSSS, DDDMMSSSSS, 0.0.. 1.393796575e+42, DDDMMSSSSS	A measure of accuracy which includes both bias squared and variance ; DDD = degrees, MM = minutes, SSSSS = thousandths of seconds, and the REAL10 value is in meters squared
Measured Image Coordinates (GDS)	REAL2D10	33	(-1.0..1.0, -1.0..1.0)	The position of the fiducial marks as measured by a comparator on a particular photograph (expressed in meters)
Merged or Layered Culture	ENUM	1	M, L	Flag indicating whether all culture LOD data in a SIF/HDI database is merged ("M") into one layer or whether the culture LOD data is layered ("L") within multiple data files
Message Classification Authority (GDS)	STR	20	--	The identification of the classification authority for the message. The code shall be in accordance with the regulations governing the appropriate security channel(s)
Message Codewords (GDS)	STR	40	--	Security compartments associated with the message.
Message Control and Handling (GDS)	STR	40	--	Security handling instructions associated with the message
Message Copy Number (GDS)	INT	5	0..99999	Copy number of the message

Field Name	Type	Length (CHARS)	Range	Description
Message Date & Time (GDS)	STR	14	DDHHMMSSZMONYY	Time (Zulu) of of origination of the message, where DD is the day of the month, HH is the hour, MM is the minute, SS is the second, the character Z, MON is the first three characters of the month, and YY is the year
Message Downgrading Event (GDS)	STR	40	--	If the Message Security Downgrade equals "999998" then this field must be present and must specify the event
Message Length (GDS)	INT	12	0..999999999999	The length in bytes of the entire message including all headers, sub-headers and data
Message Number of Copies (GDS)	INT	5	0..99999	Total number of copies of the message
Message Part Type (GDS)	STR	2	"IM", "TM"	Given as "IM" to identify the sub-header as an image sub-header, given as "TM" to identify the sub-header as a terrain sub-header.
Message Releasing Instructions (GDS)	STR	40	--	A list of countries and/or groups of countries to which the data are authorized for release
Message Security Classification (GDS)	ENUM	1	T, S, C, R, U	Classification of the entire message, where T = Top Secret, S = Secret, C = Confidential, R = Restricted, U = Unclassified
Message Security Control Number (GDS)	STR	20	--	Security control number associated with the message
Message Security Downgrade (GDS)	STR	6	--	An indicator which designates the point in time at which a declassification or downgrading action is to take place

Field Name	Type	Length (CHARS)	Range	Description
Message Title (GDS)	STR	80	--	Title of message
Message Type & Version (GDS)	STR	9	NITFNN.NN	A character string which indicates this message is using version NN.NN of NITF
Microdescriptor Type	ENUM	6	HOM_AR, DRAIN, PDIST, TRANS, VEGET, WX_FX, SN_FX, TOD, GND, ALTATT, VCOMP, TMP_FX	Designates the type of microdescriptor (HOMOGENOUS_AREA, DRAINAGE, PATTERN_-DISTRIBUTION, TRANSPORTATION, VEGETATION, WEATHER_EFFECTS, SEASONAL_EFFECTS, TIME_OF_DAY, GROUND_CONDITIONS, ALTERNATE_-ATTRIBUTES, VERTICALLY_COMPOSITE, TEMPORAL_EFFECTS)
Microdescriptor Value	STR	80	--	The value of a microdescriptor within a SIF culture tile
Mirror	(BOOLEAN; BOOLEAN; BOOLEAN; BOOLEAN)	7	(T, F; T, F; T, F; T, F)	Flag indicating whether a texture map can be mirrored along the left, right, top, and bottom edges
Model Based Texture Reference Table File Name	STR	17	--	The name of the model based texture reference table file included with SIF/HDI models
Model Centroid	REAL2D6	25	(-1.9342 e+25.. 1.9342 e+25, -1.9342 e+25.. 1.9342 e+25)	Center of a model object, expressed in meters
	REAL3D6	38	(-1.9342 e+25.. 1.9342 e+25, -1.9342 e+25.. 1.9342 e+25, -1.9342 e+25.. 1.9342 e+25)	

Field Name	Type	Length (CHARS)	Range	Description
Model Data File Name	STR	17	--	The name of the model data file included with SIF/HDI models
Model Description	STR	80	--	Description of a model
Model Form	ENUM	4	POLY, CSG, BOTH	Designator indicating the model representations present for a given model (POLYGONAL_ONLY, CSG_ONLY, CSG_AND_POLYGONAL)
Model Lat/Long	INT2D	21	(0..2147483647, 0..2147483647)	Ground coordinates of a model expressed in ten thousandths of an arc second, relative to the southwest corner of the culture tile
Model Library Type (GDS)	ENUM	15	TWO_D_STATIC, THREE_D_STATIC, THREE_D_DYNAMIC	The ID of a model library
Model Library Type	ENUM	3	2DS, 3DS, 3DD	The ID of a model library (TWO_D_STATIC, THREE_D_STATIC, THREE_D_DYNAMIC)
Model LOD	ENUM	2	L0, L1, L2, L3, L4, L5, L6, L7, L8	The level of detail of a model
Model Name (BOTH)	STR	65	--	Name of a model
Model Number (BOTH)	INT	10	0..2147483647	A unique ID number assigned to a model within a SIF database

Field Name	Type	Length (CHARS)	Range	Description
Model Reference Point	REAL2D	25	(-1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25)	A point on a 2D model which corresponds to the origin of the texture being mapped
	REAL3D	38	(-1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25)	A point on a 3D model which corresponds to the origin of the texture being mapped
Model Reference Table File Name	STR	17	--	The name of the model reference table file included with SIF/HDI culture data
Model Reference Table Index	INT	10	0..2147483647	A pointer to a model reference in a model reference table
Model String Count	INT	10	0..2147483647	The count of ASCII strings that define a SIF/DP model
Model Tie Point ID (GDS)	INT	10	0..2147483647	A unique identifier of a model tie point
Model Vertex Limit	INT	5	0..15000	The maximum number of vertices allowed in a model. 0 represents that no limit is specified
Model View Description (GDS)	STR	60	--	Textual description of the view of a model presented within an image, i.e., "Right Side of Truck"
Modified Specific Texture Flag (GDS)	BOOLEAN	5	T, F	Flag indicating whether a specific texture has been modified with synthetic data

Field Name	Type	Length (CHARS)	Range	Description
Monitor Type	ENUM	4	NOMON, LONGIT, TRANSV, MODIF	Code indicating type of raised portion of roof (NO_MONITOR, LONGITUDINAL, TRANSVERSE, MODIFIED)
NE Corner	STR	24	HDDMMSSSSSb HDDDMMSSSSSS	The northeast corner of the coverage contained within a SIF database, a culture tile, terrain tile, or specific areal texture, where HDDMMSSSSSS = hemisphere (N,S), degrees, minutes and ten thousandths of seconds of latitude; b = blank (" "); and HDDDMMSSSSSS = hemisphere (E,W), degrees, minutes and ten thousandths of seconds of longitude
New Data Flag	BOOLEAN	1	T, F	Flag indicating whether a model, culture tile, terrain tile, or texture is new to the SSDB or originally from the SSDB
Next Feature Number	INT	10	0..2147483647	If a feature is split between two culture tiles, this identifies the feature number in the neighboring culture tile; used by feature continuations and LOD cross-references
Next Manuscript ID	STR	25	--	Name of the neighboring culture tile; used by feature continuations and LOD cross-references
NITF Header Length (GDS)	INT	6	000000..276380	The length in bytes of the NITF header
Noise Removal Flag (GDS)	BOOLEAN	5	T, F	Flag indicating whether noise removal operations have been performed on the image

Field Name	Type	Length (CHARS)	Range	Description
Normal List Position	INT	5	0..32767	The pointer to a (X, Y, Z) triplet in a vertex table that specifies the normal of a polygon or a vertex
North Tile Neighbor ID (GDS)	INT	10	0..2147483647	The identifier of the neighboring areal specific texture to the north; used only for Stage 3 areal textures
Number of 2D Static Models	INT	10	0..2147483647	The total number of two-dimensional static models that are included in a SIF database
Number of 3D Static Models	INT	10	0..2147483647	The total number of three-dimensional static models that are included in a SIF database
Number of 3D Dynamic Models	INT	10	0..2147483647	The total number of three-dimensional dynamic models that are included in a SIF database
Number of Accuracy Regions	INT	2	0..20	Count of the number of accuracy regions within a tile
Number of Aggregate Features	INT	10	0..2147483647	Total number of aggregate features referenced by a superfeature
Number of Areal Features	INT	10	0..2147483647	Total number of areal features contained within a culture tile
Number of Audio Segments (GDS)	INT	3	000	Not currently supported within NITF, therefore, this value is always 0
Number of Bands (GDS)	INT	1	1..9	The number of bands of image data in the message. Used for color imagery, pseudocolor or multispectral images. The sequence of bands shall be determined by examining the Band Image Type Field. For single band images, the Number of Bands shall be 1

Field Name	Type	Length (CHARS)	Range	Description
Number of Bits Per Pixel Per Band (GDS)	INT	2	01..64	The number of data bits for each pixel for each band in the original image before compression. For multi-band images treated as a single image, the number of bits per pixel is identical for each band. 1, 8, 16 - standard for visual textures 56 - standard for SMC/FDC texture, 16 and 24 - standard for terrain.
Number of Blocks Per Column (GDS)	INT	4	1-9999	The number of image blocks in a column in the vertical direction. (P2851 has relaxed the NITF restriction of one block per column)
Number of Blocks Per Row (GDS)	INT	4	1-9999	The number of image blocks in a row or line in the horizontal direction. (P2851 has relaxed the NITF restriction of one block per row)
Number of Boundaries (GDS)	INT	5	0..32767	The total number of outlines required to specify the coverage or "footprint" of the area being transmitted
Number of Boundary Points (BOTH)	INT	5	0..32767	The total number of coordinates that define an outline around an area of data that is being transmitted
Number of Child Features	INT	10	0..2147483647	Total number of children features referenced by a superfeature
Number of Child Superfeatures	INT	10	0..2147483647	Total number of children superfeatures referenced by a superfeature
Number of Clusters	INT	4	0..1000	Number of clusters in a model
Number of Collision Test Points	INT	5	0..15000	Number of collision test points in a model

Field Name	Type	Length (CHARS)	Range	Description
Number of Colors	INT	10	0..2147483647	Number of colors in a color table
Number of Component Texture References	INT	5	0..65535	Number of textures referenced by a model component
Number of Components	INT	4	0..1000	Number of components in a model
Number of Control Points (GDS)	INT	5	0..32767	The total number of control points
Number of Coordinate Pairs	INT	5	0..20000	The total number of coordinates contained within a segment
Number of Cross Sections	INT	3	0..100	Number of 2D cross section in a CSG model
Number of Culture Files	INT	5	0..32767	The total number of SIF/DP culture files contained within a transmittal
Number of Culture Segments	INT	10	0..2147483647	The total number of culture segments that are referenced by a given feature
Number of Culture Tiles	INT	10	0..2147483647	The total number of culture tiles included within a SIF database
Number of Data Sources (BOTH)	INT	4	0..1000	Number of data sources in a data source table
Number of Database Boundary Coordinates	INT	5	0..32767	Number of coordinates that comprise the outline around all of the culture tiles in the culture database
Number of Edges	INT	5	0..20000	Number of edges in a model, a cluster, a component, or a polygon
Number of Embedded Higher-Resolution Islands	INT	5	0..32767	The total number of high resolution islands within a culture tile

Field Name	Type	Length (CHARS)	Range	Description
Number of Enumerated Items	INT	5	0..32767	Number of values for a user-defined FACS attribute that is an enumerated type
Number of FACS Attributes for This Entry	INT	5	0..32767	Number of FACS attributes for a specific FACS table entry
Number of FACS List Pointers	INT	5	0..10000	Number of pointers to a FACS Table
Number of FACS Records for This Entry	INT	5	0..10000	Number of FACS attributes in a FACS table entry
Number of FACS Table Entries	INT	10	0..2147483647	Number of entries in a FACS table
Number of Feature Continuations	INT	10	0..2147483647	Total number of feature continuation records associated with a feature
Number of FID/FDC Cross-References	INT	10	0..2147483647	Number of entries in a FID/FDC cross reference table
Number of Fiducial Coordinates (GDS)	INT	5	0..32767	Total number of fiducial coordinates associated with an image
Number of Generic Textures	INT	10	0..2147483647	The total number of generic texture maps included within a SIF database
Number of Generic Textures In Set (BOTH)	INT	10	0..2147483647	The number of generic textures included within a generic texture set
Number of Generic Texture Sets (GDS)	INT	10	0..2147483647	The total number of generic texture sets in a SIF data base, where a generic texture set is a set of generic textures that represent the same entity, and each member of the set has a different size and/or resolution

Field Name	Type	Length (CHARS)	Range	Description
Number of Geographic Tie Points (GDS)	INT	10	0..2147483647	The total number of tie points for a SIF data base or for a single areal texture
Number of Higher LOD Cross References	INT	5	0..32767	The total number of higher LOD cross references associated with a culture feature
Number of IGES Records	INT	10	0..2147483647	Total number of IGES records used in describing the CSG representation of a model
Number of Image Comments (GDS)	INT	1	0..9	The number of free form image comments
Number of Images (GDS)	INT	3	000..999	The number of separate images included in the message
Number of Instances	INT	3	0..127	The total number of model references associated with a feature
Number of Island Boundary Coordinates	INT	5	0..32767	The total number of coordinates that define an outline around an area of higher resolution data within a culture tile
Number of Labels (GDS)	INT	3	000..999	The number of separate labels included in the message. This value will always be '000' for SIF
Number Of Lights	INT	5	0..32767	Number of lights in a point light string
Number of Linear Features	INT	10	0..2147483647	The total number of linear features contained within a culture tile
Number of LODs	INT	1	0..6	Number of different LODs that are contained within a culture database

Field Name	Type	Length (CHARS)	Range	Description
Number of Lower LOD Cross References	INT	1	0..1	The total number of lower LOD cross references associated with a culture feature
Number of LUT Entries (GDS)	INT	5	00001..65536	The number of entries in each of the look-up tables for a band of an image
Number of LUTs (GDS)	INT	1	0..4	The number of look-up tables used in displaying a band of an image
Number of Manuscript Boundary Coordinates	INT	3	0..127	The total number of coordinates used to define the boundary of a culture tile
Number of Microdescriptors	INT	5	0..32767	Number of microdescriptors associated with a culture feature, a model polygon or a model component
Number of Model LOD Texture References	INT	5	0..65535	Number of textures referenced by a model LOD
Number of Model LODs	INT	1	0..9	Number of levels of detail of a model
Number of Model References	INT	10	0..2147483647	Number of models referenced by features contained in a culture tile
Number of Model Tie Points (GDS)	INT	10	0..2147483647	The total number of tie points for a SIF data base or for a single model texture
Number of Model Vertices	INT	10	0..2147483647	Number of model vertices for all LODs of a model
Number of Models	INT	10	0..2147483647	Number of models contained in a model library
Number of Models in Image (GDS)	INT	3	0..999	The number of models that are represented in some manner within an image

Field Name	Type	Length (CHARS)	Range	Description
Number of Non-Static Presentation Information Files (GDS)	INT	3	000	The number of non-static presentation information files included in the message. This value will always be 000 for SIF
Number of Parent Superfeatures	INT	10	0..2147483647	Total number of superfeatures that reference a given superfeature
Number of Pixels Per Block Horizontal (GDS)	INT	5	0001..99999	The number of pixels horizontally in each block
Number of Pixels Per Block Vertical (GDS)	INT	5	0001..99999	The number of pixels vertically in each block
Number of Point Features	INT	10	0..2147483647	The total number of point features contained in a culture tile
Number of Point Light Features	INT	10	0..2147483647	The total number of point light features contained in a culture tile
Number of Point Light Strings	INT	5	0..65535	Number of point light strings contained in a model
	INT	10	0..2147483647	Number of point light strings contained in a culture tile
Number of Polygon Texture References	INT	5	0..65535	Number of textures referenced by a polygon in a model
Number of Polygonization Instructions	INT	5	0..65535	Number of instructions for polygonizing a CSG model
Number of Polygons	INT	5	0..10000	Number of polygons in a model, a model component or a cluster
Number of Polygons Along Surface 1	INT	5	0..10000	Number of polygons to be generated along surface 1 when polygonizing a CSG model

Field Name	Type	Length (CHARS)	Range	Description
Number of Polygons Along Surface 2	INT	5	0..10000	Number of polygons to be generated along surface 2 when polygonizing a CSG model
Number of Segment Backpointers	INT	5	0..32767	Total number of references from a segment back to the feature(s) that reference the segment
Number of Sensors (GDS)	INT	3	0..127	Number of sensors used to form a composite processed image
Number of Separation Planes	INT	3	0..127	Number of separation planes in a model that divide the model into a number of clusters
Number of SMC/FDC Textures	INT	10	0..2147483647	The total number of SMC/FDC Texture maps included within a SIF database
Number of Sources	INT	5	0..32767	The total number of different sources used to compile a SIF/DP culture data tile
Number of Stage 1 Specific Areal Textures	INT	10	0..2147483647	The total number of Stage 1 Specific Areal Texture maps included within a SIF database
Number of Stage 1 Specific Model Textures	INT	10	0..2147483647	The total number of Stage 1 Specific Model Texture maps included within a SIF database
Number of Stage 2 Specific Areal Textures	INT	10	0..2147483647	The total number of Stage 2 Specific Areal Texture maps included within a SIF database
Number of Stage 2 Specific Model Textures	INT	10	0..2147483647	The total number of Stage 2 Specific Model Texture maps included within a SIF database

Field Name	Type	Length (CHARS)	Range	Description
Number of Stage 3 Specific Areal Textures	INT	10	0..2147483647	The total number of Stage 3 Specific Areal Texture maps included within a SIF database
Number of Stage 3 Specific Model Textures	INT	10	0..2147483647	The total number of Stage 3 Specific Model Texture maps included within a SIF database
Number of Stereo Mates (GDS)	INT	3	0..127	Number of images that overlap a given image
Number of Structures	INT	10	0..2147483647	Total number of structures for a feature per square kilometer
Number of Subsidiary Model References	INT	4	0..1000	Number of subsidiary models referenced by a model as a component
Number of Substituted Features	INT	5	0..32767	Total number of features that reference a specific model
Number of Symbols (GDS)	INT	3	000..999	The number of separate symbols included in the message. This value will always be '000' for SIF
Number of Terrain Comments (GDS)	INT	1	0..9	The number of free form terrain comments
Number of Terrain Files (GDS)	INT	3	000..999	The total number of terrain files that are being transmitted
Number of Terrain Tiles	INT	10	0..2147483647	The total number of terrain tiles included within a SIF database
Number of Texels Per Column	INT	10	0..2147483647	The total number of texels per column contained in a SIF texture map
Number of Texels Per Row	INT	10	0..2147483647	The total number of texels per row contained in a SIF texture map

Field Name	Type	Length (CHARS)	Range	Description
Number of Text Files (GDS)	INT	3	000.999	The number of separate text files that are included in the message. This value will always be '000' for SIF
Number of Texture Pattern Coordinates	INT	5	0..65535	Total number of coordinates associated with a texture; used to warp the texture onto a model or terrain using vertex to vertex mapping techniques; the number should be the same as the number of vertices of the polygon being mapped
Number of Texture References	INT	10	0..2147483647	Number of entries in a texture reference table The total number of texture references contained in a culture tile
Number of Tie Point References (GDS)	INT	10	0..2147483647	The number of texture maps that share a specific tie point
Number of Tiles	INT	5	0..32767	The total number of tiles contained within a culture database
Number of User-defined FACS Attribute Codes	INT	5	0..32767	Number of user-defined FACS attributes contained in the user defined FACS Table
Number of Vertex Colors	INT	5	0..15000	Number of vertex colors associated with a polygon in a model
Number of Vertex Normals	INT	5	0..15000	Number of vertex normals associated with a polygon in a model
Number of Vertices	INT	5	0..15000	Number of vertices in a model LOD, a polygon, a model component, or a cluster

Field Name	Type	Length (CHARS)	Range	Description
Object Or Material Texture Flag (GDS)	ENUM	8	OBJECT, MATERIAL	Flag indicating whether a generic texture is applied to a certain object or if it is representative of a material (e.g., generic texture for a certain tree or a road would be classified as OBJECT; texture for tree bark or asphalt would be classified as MATERIAL)
Object Volume	REAL6	12	0.0..1.93428 e+25	The internal volume of an object, in liters
Occlusion Removal Flag (GDS)	BOOLEAN	5	T, F	A flag indicating whether occluding objects have been removed from an image
Offset Vector	INT2D	21	(-324000000.. 324000000; -648000000, 648000000)	Placement information for a model reference expressed in thousandths of seconds relative to southwest corner. This vector is used to determine the location of a model based on its placement point in relation to the first coordinate of the referencing feature
Omega (GDS)	REAL6	12	0.0..360.0	A rotation angle around the x-axis. A positive angle rotates the y-axis toward the z-axis (expressed in degrees)
Orientation	REAL10	16	0.0..360.0	Orientation of a point, point light , or point light string feature from due north, expressed in degrees
Orientation Angle	REAL10	16	0.0..360.0	The orientation of a model from due north in the clockwise direction, expressed in degrees

Field Name	Type	Length (CHARS)	Range	Description
Orientation Vectors	(REAL2D10, REAL2D10)	67	-1.39379657e+42.. 1.39379657e+42, -1.39379657e+42.. 1.39379657e+42; -1.39379657e+42.. 1.39379657e+42, -1.39379657e+42.. 1.39379657e+42	For 2D models or 2D culture, two vectors aligning the x and the y axes of a texture to be mapped with 2D model space or world space; used for mapping onto a model using model-based texturing or onto culture using global-based texturing
	(REAL3D10, REAL3D10)	101	-1.39379657e+42.. 1.39379657e+42, -1.39379657e+42.. 1.39379657e+42, -1.39379657e+42.. 1.39379657e+42; -1.39379657e+42.. 1.39379657e+42, -1.39379657e+42.. 1.39379657e+42, -1.39379657e+42.. 1.39379657e+42	For 3D models, 3D culture, or 3D terrain polygons, two vectors aligning the x and the y axes of a texture to be mapped with 3D model space or 3D world space; used for mapping onto a model using model-based texturing or onto terrain/culture using global-based texturing
Original Data Sources Used (GDS)	STR	80	--	A list of the original sources used to create an image
Originating Station ID (GDS)	STR	10	--	Identification code of the originating system (terrain or texture)
Originator	STR	80	--	Name (organization) of the creator of a SIF tape
Originator's Name (GDS)	STR	27	--	Name of the operator who originated the message
Originator's Phone Number (GDS)	STR	18	--	Phone number of the operator who originated the message
P2851 Binary Separation Planes Flag	BOOLEAN	1	T, F	Indicates whether the separation planes are defined based on P2851 definition

Field Name	Type	Length (CHARS)	Range	Description
Percent of Cloud Cover (GDS)	INT	3	0..100	Percentage of the image which is covered by clouds
Percent of Roof Coverage	REAL6	12	0.0..100.0	Percentage of the amount of roof coverage for an areal feature
Percent of Shadow Cover (GDS)	INT	3	0..100	Percentage of the image which is covered by shadows
Percent of Specific Texture (GDS)	INT	3	0..100	Percentage of an image that is specific to the geographic location that it is being placed at (i.e., the percentage of texture that has not been replaced by generic or "best-guess" texture)
Percent of Texture in Tile (GDS)	INT	3	0..100	Percentage of a Stage 3 texture tile that has been filled with actual texture (i.e., some void areas may exist within a texture tile)
Percent of Tree Coverage	REAL6	12	0.0..100.0	Percentage of the amount of tree coverage for an areal feature
Phi (GDS)	REAL6	12	0.0..360.0	A rotation angle around the y-axis. A positive angle rotates the z-axis toward the x-axis (expressed in degrees)
Placement Point	REAL2D6	25	(-1.93428 e+25.. 1.93428 e+25, -1.93428 e+25.. 1.93428 e+25)	Point within a model used for easy placement of the model
	REAL3D6	38	(-1.93428 e+25.. 1.93428 e+25, -1.93428 e+25.. 1.93428 e+25, -1.93428 e+25.. 1.93428 e+25)	

Field Name	Type	Length (CHARS)	Range	Description
Point Light Position	REAL2D6	25	(-1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25)	Position (x,y or x,y,z) in meters of a point light within a point light string in a model
	REAL3D6	38	(-1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25)	
Point Light String Delta	INT3D	35	(-2147483648.. 2147483647; -2147483648.. 2147483647; -2147483648.. 2147483647)	Distance between lights in a point light string, expressed as a vector using relative coordinates with elevation in millimeters
Point Light String Origin	INT3D	35	(-2147483648.. 2147483647; -2147483648.. 2147483647; -2147483648.. 2147483647)	Location of the first light in the point light string relative to the southwest corner of the tile in ten-thousandths of a second with elevation in millimeters
Polygon Alignment Vector	REAL2D6	25	(-1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25)	Vector lying on the plane of the polygon and aligned in the direction of the x-axis of the texture map before the Rotation About Texture Origin value is applied; the type is determined by the dimension of the polygon
	REAL3D6	38	(-1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25)	

Field Name	Type	Length (CHARS)	Range	Description
Polygon ID	INT	10	0..2147483647	Unique identifier of a polygon within a model
Polygon Illumination Type	ENUM	5	SELF, SUN, NOSUN	Indicator identifying how illumination of this polygon is to be computed (SELF_LUMINOUS, SUN_LUMINOUS, NO_SUN_ILLUMINATION)
Polygon Landing Light Illumination	BOOLEAN	1	T, F	Flag indicating whether this feature gets illuminated by aircraft landing lights
Polygon Non-Occulting	BOOLEAN	1	T, F	Indicator that the color of the polygon is additive to the background color
Polygon Non-Shadow	INT	5	0..32767	The amount of shadow a polygon presents when illuminated or irradiated
Polygon Normal	REAL3D6	38	(-1.0..1.0, -1.0..1.0, -1.0..1.0)	The normalized vector perpendicular to a polygon
Polygon Reference Point	REAL2D6	25	(-1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25)	A point on a 2D polygon which corresponds to the origin of the texture being mapped
	REAL3D6	38	(-1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25)	A point on a 3D polygon which corresponds to the origin of the texture being mapped
Positional Accuracy Standards (GDS)	STR	80	--	Description of standard used in geopositioning texture

Field Name	Type	Length (CHARS)	Range	Description
Predominant Height	REAL10	16	-1.393796575e+42 .. 1.393796575e+42	Height of feature above terrain, largest vertical axis, expressed in centimeters
Processing Comments (GDS)	STR	80	--	Textual comments on any special processing performed on the texture that may be helpful for later use and evaluation of the texture
Producer Code	STR	8	CCAAABBB	Country, Agency, Branch of the producer of the data, DIA country codes used for the first two characters
Radius	REAL6	12	0.0..1.9342 e+25	Radius of encompassing sphere around a model in meters.
	REAL10	16	0.0.. 1.393796575e+42	Radius of encompassing sphere around a feature in meters.
Recipient	STR	80	--	Name (organization) of the recipient of a SIF tape

Field Name	Type	Length (CHARS)	Range	Description
Record Keyword	STR	2	"2L", "2S", "3D", "3L", "3S", "A1", "A2", "A3", "AF", "AR", "CE", "CH", "CR", "CS", "CT", "DD", "DH", "DL", "DR", "DS", "FB", "FC", "FE", "FP", "FT", "FX", "GB", "GX", "HL", "IS", "LF", "LH", "LL", "LR", "LS", "M1", "M2", "M3", "MB", "ME", "MH", "MI", "ML", "MR", "MT", "MX", "NH", "NM", "NX", "PF", "PI", "PL", "PM", "PO", "PR", "RE", "RT", "SB", "SE", "SF", "SH", "SP", "TD", "TE", "TH", "TP", "TX", "UE", "UF", "VB", "VC", "VN", "VP", "VX"	Two character sentinel that precedes a SIF/HDI Record
Rectification (GDS)	STR	20	"RECTIFIED", "EPIPOLAR", "NONE", etc.	Definition of the type of rectification process used on an image
Referenced Model Library Type	ENUM	3	2DS, 3DS, 3DD	The model library that a referenced model belongs to (TWO_D_STATIC, THREE_D_STATIC, THREE_D_DYNAMIC)

Field Name	Type	Length (CHARS)	Range	Description
Referenced Model LOD	ENUM	2	L0, L1, L2, L3, L4, L5, L6, L7, L8	The referenced level of detail of a referenced model
Referenced Model Number	INT	10	0..2147483647	The model ID number of a referenced model
Reflectance	REAL6	12	0.0..1.0	Ratio of radiant energy reflected by an object to the amount incident upon it
Relative Coordinate	INT2D	23	(-2147483648.. 2147483647; -2147483648.. 2147483647)	Coordinate value in ten-thousandths of a second realtive to southwest corner; used for accuracy regions
Relative Coordinates (GDS)	REAL3D6	38	(-1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25)	Model coordinates (x,y,z) used to tie a texture to a model; each coordinate corresponds to an Image Coordinate; if the model is 2D, then z = 0.0; else model is 3D, then z is in meters
Relative Horizontal Accuracy	STR	16	--	Definition of horizontal accuracy standard applying to a data source product
Relative Latitude	BINARY INT	4	-2147483648.. 2147483647	Latitudinal value of a coordinate relative to the southwest corner of a culture tile, in ten thousandths of an arc second; used for culture coordinates

Field Name	Type	Length (CHARS)	Range	Description
Relative Longitude	BINARY INT	4	-2147483648.. 2147483647	Longitudinal value of a coordinate relative to the southwest corner of a culture tile, in ten thousandths of an arc second; used for culture coordinates
Relative Vertical Accuracy	STR	16	--	Definition of vertical accuracy standard applying to a data source product
Releasing Instructions	STR	40	--	A list of countries and/or groups of countries to which the SIF database or elements within the SIF database are authorized for release.
Reliability of Data (BOTH)	INT	3	0..100	The degree of reliability of the data
RGB/HCV Color Value	INT3D	17	(0..32767, 0..32767, 0..32767)	Hue, Chroma, Value value used to specify the color of an object. Hue - 0 (0 degrees) for blue, 10922 (120 degrees) for red, 21845 (240 degrees) for green.
	INT3D	11	(0..255, 0..255, 0..255)	Red, Green, Blue value used to specify the color of an object Format is determined by the Color Definition Type value
Right Tile Neighbor ID (GDS)	INT	10	0..2147483647	The identifier of the neighboring model specific image to the right of the current image; used only for Stage 3 model textures

Field Name	Type	Length (CHARS)	Range	Description
Roof Type	ENUM	4	FLAT, SHED, DECK, GABL, HIPP, GAMB, MANS, SAWT, CURV, CONI, NO	The type of roof associated with a structure (FLAT, SHED, DECK, GABLE, HIPPED, GAMBREL, MANSARD, SAWTOOTH, CURVED, CONICAL, NONE)
Rotation Angles	REAL3D6	38	(0.0..360.0; 0.0..360.0; 0.0..360.0)	Rotation angles about a subsidiary model's x, y, and z axes in its local coordinate system (right-handed)
Rotation About Texture Origin	REAL6	12	0.0..360.0	Rotation angle of a texture when applying it to a polygon; rotation is applied after the texture has been placed on the polygon using the other mapping parameters
Scale Factor	REAL3D6	38	(-1.9342 e+25.. 1.9342 e+25; -1.9342 e+25.. 1.9342 e+25; -1.9342 e+25.. 1.9342 e+25)	X, Y, and Z scale factors used on a subsidiary model when associating it with another model
Scan Filter ID (GDS)	ENUM	5	MONO RED GREEN BLUE	Filter used when scanning a hardcopy image
Scan Resolution (GDS)	STR	10	--	The distance which is represented by a pixel in image space dimensions. The range is 0.1 micron to 1 mm in image dimensions
Scanner ID (GDS)	STR	20	--	Identifier or name of the device used to scan an image
Section Identifier	STR	80	--	Alphanumeric string identifying the SIF data section where an ASCII SIF file is found; usually found in the first record of a SIF ASCII file

Field Name	Type	Length (CHARS)	Range	Description
Security Classification	ENUM	1	T, S, C, R, U	Classification of a SIF database or elements within a SIF/HDI database, where T = Top Secret, S = Secret, C = Confidential, R = Restricted, U = Unclassified; derived from an NITF type and left in this form for consistency with NITF
Security Control Number	STR	20	--	Security control numbers associated with the SIF database or elements within a SIF database. The format is in accordance with the regulations governing the appropriate security channel(s)
Security Downgrade	STR	6	YYMMDD	The date that a SIF database or elements within a SIF database were downgraded, where YY = year, MM = month, and DD = day
Security Level	STR	2	--	Code identifying the level of classification
Segment Direction	ENUM	3	CW, CC, DCW, DCC, ICW, ICC	Direction of traversal of a feature segment within a feature (CLOCKWISE, COUNTER_CLOCKWISE, DISJOINT_CLOCKWISE, DISJOINT_COUNTER_CLOCKWISE, INSIDE_CLOCKWISE, INSIDE_COUNTER_CLOCKWISE)
Segment File Name	STR	17	--	The name of the segment file included with SIF/HDI culture data
Segment ID Number	INT	10	1..2147483647	Unique identifier of a culture segment within a culture tile

Field Name	Type	Length (CHARS)	Range	Description
Self-Emitter	BOOLEAN	1	T, F	Indicates that an object has self heating characteristics
Sensor ID (GDS)	INT	10	0..2147483647	A unique identifier of a sensor within a SIF texture transmittal
Sensor Name (GDS)	STR	20	--	Name of the sensor used to capture the image, e.g., "LANDSAT-2 RBV" or "MINOLTA 7S-II"
Sensor Type (GDS)	ENUM	18	FRAME, MECHANICAL_ SCANNER, PANORAMIC, PUSHBROOM, STRIP, OTHER	Type of sensor used to capture the image (e.g., any ordinary camera or a metric camera would be FRAME, SPOT satellite would be PUSHBROOM, and LANDSAT satellite would be MECHANICAL_SCANNER)
Sensor Types Supported	ENUM	3	VIS, IR, RAD	Flags indicating support for different types of simulators (VISUAL, INFRARED, or RADAR)
Sensors Supported	BOOLEAN; BOOLEAN; BOOLEAN	5	(T, F; T, F; T, F)	Flags indicating support for different types of simulators (radar, visual, infrared, respectively)
Separation Plane Coefficients	REAL4D6	51	(-1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25)	The A, B, C, D coefficients of the equation of a separation plane that divides a model into clusters
Separation Plane Number	INT	5	0..32767	The unique ID number assigned to a separation plane

Field Name	Type	Length (CHARS)	Range	Description
Sequence Number	INT	5	0..65535	A unique sequencing identifier of an IGES record
Shading Type	ENUM	6	FIXED, FLAT, SMOOTH	Name of the shading method used to shade a model
Shadow Minimization Flag (GDS)	BOOLEAN	5	T, F	Flag indicating whether operations to minimize the effects of shadows within an image have been performed
Shape Code	ENUM	7	RP, HEMIS, PYRAMID, CONE, CYL, OTH	Code indicating the shape of a feature (RECTANGULAR_PARALLELEPIPED, SPHERE_HEMISPHERE, PYRAMID, CONE, CYLINDER, OTHER)
Shared Segment Flag	BOOLEAN	1	T, F	Flag indicating whether a culture segment is referenced by more than one feature
SIF Format	ENUM	7	SIF_HDI, SIF_DP	Designator indicating one of the two major SIF formats
SIF Version Number	STR	5	--	Released version number for SIF document used in preparing a SIF database
SIF/HDI Sentinel	STR	7	"SIF/HDI", "SIF/END"	Textual flag indicating the start or end of a SIF/HDI record within the NITF record structure
Source Agency/Project (BOTH)	STR	16	--	Name of the agency or project that created the digital source, e.g., "SOFATS", "P2851", etc.
Source Date (BOTH)	STR	6	YYMMDD	Date the digital source was created, where YY = Year, MM = Month, DD = Day
Source ID Number (BOTH)	INT	5	0..32767	Unique identifier of an entry in the data source table

Field Name	Type	Length (CHARS)	Range	Description
Source Name (BOTH)	STR	20	--	Name of the original source, e.g., "EOSAT", "General Electric", etc
Source Originator	STR	20	--	Name of the agency or project that created the digital source, e.g., "SOFATS", "P2851", etc.
Source Simulator	STR	4	--	Designator for the particular simulator for which a model was created, if the model is not generic
Source Type (BOTH)	ENUM	1	H, S	Flag indicating hardcopy (H) or softcopy (S) source
South Tile Neighbor ID (GDS)	INT	10	0..2147483647	The identifier of the neighboring areal specific image to the south; used only for Stage 3 areal textures
Special Environmental Conditions (GDS)	STR	80	--	Textual description about any special conditions present when an image was captured
Specular	BOOLEAN	1	T, F	Flag indicating that the polygon has the quality of being mirror-like
SSDB LOD Number	ENUM	2	L0, L1, L2, L3, L4, L5	Level of detail within the SSDB that a culture tile or high resolution islands within the tile are to reside at, where L0 approximates 300 meter resolution, L1 approximates 100 meter resolution, L2 approximates 30 meter resolution, L3 approximates 10 meter resolution, L4 approximates 3 meter resolution, and L5 approximates 1 meter resolution

Field Name	Type	Length (CHARS)	Range	Description
SSDB Texture ID (GDS)	INT	10	0..2147483647	ID number assigned to a texture; it is unique within a P2851 SSDB texture library; it corresponds to a positive integer which is equal to the original SSDB texture ID if the texture originated from the SSDB; this value is zero if the image has never been stored within the SSDB
Standard Image Filter Code (GDS)	STR	3	--	This field is reserved for future use
Standard Terrain Filter Code	STR	3	--	This field is reserved for future use
Substituted Feature Number	INT	10	0..2147483647	Identifier of a feature that is replaced by a model reference
Sun Azimuth (GDS)	REAL6	12	0.0..360.0	The clockwise angle measured in the horizontal plane, at the observer, between due north and the vertical projection of the center of the sun onto the horizon (expressed in degrees)
Sun Elevation (GDS)	REAL6	12	-90.0..90.0	The angle measured in a vertical plane, at the observer, between the horizon and the center of the sun, where negative values are below the horizon (expressed in degrees)
Superfeature Description	STR	160	--	Textual description of a superfeature
Superfeature File Name	STR	17	--	The name of the superfeature file included with SIF/HDI culture data
Superfeature ID	INT	10	0..2147483647	The unique identifier for a superfeature category

Field Name	Type	Length (CHARS)	Range	Description
Surface Material Category	INT	3	0..255	DMA surface material category code with room for expansion
Surface Material Subtype	INT	3	0..255	Indicator to further refine DMA SMCs, used to add spatial and/or temporal breakup characteristics
SW Corner	STR	24	HDDMMSSSSSb HDDDMMSSSSSS	The southwest corner of the coverage contained within a SIF database, a culture tile, terrain tile, or specific areal texture, where HDDMMSSSSSS = hemisphere (N,S), degrees, minutes and ten thousandths of seconds of latitude; b = blank (" "); and HDDDMMSSSSSS = hemisphere (E,W), degrees, minutes and ten thousandths of seconds of longitude
Synthetic Data Flag (GDS)	BOOLEAN	5	TRUE, FALSE	Flag indicating whether a piece of data is real or not
Synthetic Data Flag	BOOLEAN	1	T, F	Flag indicating whether a piece of data is real or not
System Type (GDS)	STR	6	--	Reserved for future use
Target ID (GDS)	STR	17	BBBBBBBBBBFFFFF CC	The identification of the target, consisting of 10 characters of Basic Encyclopedia (BE), 5 characters of functional category code, and the 2 character country code as specified by FIPS-PUB 10-3
Terrain Classification Authority (GDS)	STR	20	--	The identity of the classification authority of the terrain map

Field Name	Type	Length (CHARS)	Range	Description
Terrain Codewords (GDS)	STR	40	--	Security compartments associated with a terrain map. Digraphs in accordance with DIAM 65-19 and its supplements, trigraphs not contained in DIAM 65-19, and complete codewords or project numbers may be used. The selection of a relevant set of codewords is implementation specific. The individual values are separated by single spaces
Terrain Comment (GDS)	STR	80	--	Field to be used for free form comments. May be used for terrain specific information. If the comment is classified, then it will be preceded by the classification, including codeword(s). Omitted if Number of Terrain Comments is zero.
Terrain Compression (GDS)	ENUM	2	NC, C0, C1, C2	If the terrain map is transmitted in a compressed form, the letter C followed by a number between 0 and 2 is used to indicate the compression scheme used (C0 = compressed with a user specified algorithm, C1 = one bit, C2 = ARIDPCM). Given as NC if the image is not compressed
Terrain Control and Handling (GDS)	STR	40	--	Security handling instructions associated with a terrain map
Terrain Coordinate System (GDS)	ENUM	1	G, O	Coordinate system of the terrain map where G = geodetic, O = Other. While NITF allows other values, P2851 has restricted the range of this field; for texture to be accepted into the active SSDB, the coordinate system must be geodetic

Field Name	Type	Length (CHARS)	Range	Description
Terrain Data File Name	STR	17	--	The name of the terrain data file included with the SIF/HDI terrain data
Terrain Downgrading Event (GDS)	STR	40	--	If the Terrain Security Downgrade equals "999998" then this field must be present and must specify the event
Terrain Feature Identifier	ENUM	6	NO, RIDGE, VALLEY, POLY, CONT, ELEV, COAST, OTR	Identifier indicating the feature as a terrain feature rather than a culture feature (NONE, RIDGELINE, VALLEY, POLYGON, CONTOUR, SPOT_ELEVATION, COASTLINE, OTHER)
Terrain Filter Condition (GDS)	ENUM	1	N	Reserved for future use
Terrain Geographic Location (GDS)	STR	91	DDMMSSSSSXb DDDMMSSSSSY (4 times)	Geographic location of the terrain data in geodetic coordinates. Geodetic coordinates are given as the latitude and longitude of the four corners in clockwise order beginning with the top left corner of the image as it is transmitted, where DDMMSSSSSX represents degrees, minutes, and thousandths of seconds of latitude with X= N or S for north or south, and DDDMMSSSSSY represents degrees, minutes, and thousandths of seconds of longitude with Y = E or W for east or west. b = blank (" "). Omitted if Terrain Coordinate System equals Other. P2851 has altered the size and accuracy to meet simulation requirements. Note: This is a composite field made up of 4 items; by definition, each item is separated by the standard intra-field separator (" ").
Terrain ID (GDS)	STR	10	--	Textual identification of the terrain map

Field Name	Type	Length (CHARS)	Range	Description
Terrain Location (GDS)	STR	10	RRRRRCCCCC	An ordered pair defining the location in cartesian coordinates where the first terrain post of the first line of the terrain map is to be located, where RRRRR is the row and CCCCC is the column where the upper left corner of the terrain map is to be located. (Not used by SIF)
Terrain Magnification (GDS)	STR	4	--	The magnification (or reduction) factor of the transmitted terrain map relative to the original source terrain map
Terrain Mode (GDS)	ENUM	1	S, I	Flag indicating band sequential "S" or band interleaved "I" transmission format. For terrain, the value will always be "S"
Terrain Releasing Instructions (GDS)	STR	40	--	A list of countries and/or groups of countries to which the data are authorized for release.
Terrain Security Classification (GDS)	ENUM	1	T, S, C, R, U	Classification of the terrain map and terrain sub-header. T = Top Secret, S = Secret, C = Confidential, R = Restricted, U = Unclassified
Terrain Security Control Number (GDS)	STR	20	--	Security control numbers associated with the terrain map. The format is in accordance with the regulations governing the appropriate security channel(s)
Terrain Security Downgrade (GDS)	STR	6	--	An indicator which designates the point in time at which a declassification or downgrading action is to take place.

Field Name	Type	Length (CHARS)	Range	Description
Terrain Source (GDS)	STR	80	--	Description of the source of the terrain map. If the source is classified, then it will be preceded by the classification, including codeword(s)
Terrain Subheader File Name	STR	17	--	The name of the terrain subheader file included with the SIF/BDI terrain data
Terrain Sync Code (GDS)	INT	1	0, 4	A field that indicates whether a synchronization code has been provided for uncompressed or ARIDPCM compressed data
Terrain Title (GDS)	STR	80	--	Title of the terrain map
Terrain Type (GDS)	STR	8	--	Reserved
Texel Value (GDS)	BINARY INT	--	--	The intensity, color, multispectral, SMC/FDC, or LUT pointer data for a single texel within a texture; the data content and length is variable between textures (but consistent within a single texture) and is determined by the values of other data fields
Texture Description (GDS)	STR	80	--	Textual description of texture
Texture ID (BOTH)	INT	10	0..2147483647	ID number assigned to a texture; unique within an SSDB texture library

Field Name	Type	Length (CHARS)	Range	Description
Texture Library (GDS)	ENUM	15	STAGE_1_AREAL_ TEXTURE, STAGE_2_AREAL_ TEXTURE, STAGE_3_AREAL_ TEXTURE, STAGE_1_MODEL_ TEXTURE, STAGE_2_MODEL_ TEXTURE, STAGE_3_MODEL_ TEXTURE, GENERIC_TEXTURE, SMC_FDC_TEXTURE	ID of one of the eight texture libraries
Texture Library	ENUM	4	S1AT, S2AT, S3AT, S1MT, S2MT, S3MT, GNRC, SMFD	ID of one of the eight texture libraries (STAGE_1_AREAL_ TEXTURE, STAGE_2_AREAL_TEXTURE, STAGE_3_AREAL_TEXTURE, STAGE_1_MODEL_TEXTURE, STAGE_2_MODEL_TEXTURE, STAGE_3_MODEL_TEXTURE, GENERIC_TEXTURE, SMC_FDC_TEXTURE
Texture Map Reflectance	REAL6	12	0.0..1.0	Reflectance value assigned to a texture map
Texture Mapping Set ID	INT	10	0..2147483647	ID number identifying a set of textures used together when mapping (e.g., a summer texture set and a winter texture set may exist)
Texture Mapping Type	ENUM	2	GB, MB, FB, VV, NM	Method used in mapping texture onto terrain, culture, and models (GLOBAL_BASED, MODEL_BASED, FACE_BASED, VERTEX_TO_VERTEX, NON_MAPPED)

Field Name	Type	Length (CHARS)	Range	Description
Texture Origin	INT2D	11	(0..99999, 0..99999)	Location designated as the origin within a texture
Texture Pattern Coordinates	INT2D	11	(0..99999, 0..99999)	Positions within an image that are to be tied to the vertices of a model polygon when performing a vertex-to-vertex texture mapping
Texture Reference Table File Name	STR	17	--	The name of the texture reference table file included with SIF/HDI culture data
Texture Reference Table Index	INT	5	0..65535	A pointer to a texture reference in a texture reference table
Texture Scale	REAL2D6	25	(0.0.. 1.93428 e+25, 0.0.. 1.93428 e+25)	Scale parameters applied to a texture map
Texture Type	ENUM	4	RGB, GRAY, MULTI, SMFD	Type of data contained within a texture map (RBG, INTENSITY, MULTI_SPECTRAL, SMC_FDC)
Three-D Coordinate File Name	STR	17	--	The name of the three-dimensional coordinate file included with SIF/HDI culture data
Tile Information File Name	STR	17	--	The name of the tile information file included with SIF/HDI culture data

Field Name	Type	Length (CHARS)	Range	Description
Translation	REAL2D6	25	(-1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25)	Translation parameters applied to a referenced model or a referenced photo texture
	REAL3D6	38	(-1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25; -1.93428 e+25.. 1.93428 e+25)	
Translucency	REAL6	12	0.0..100.0	The degree to which a surface is transparent
Transmittal ID	STR	10	YYMMDDOOXX	Unique identifier of a SIF tape transmittal where YY = year, MM= month, DD = day of tape creation; OO = unique originator's code supplied by P2851 facility; and XX = sequence number for transmittals by the originator on that day (e.g., 9206152301 would be used for the first SIF tape transmittal created on 15 June 1992 by orignator 23)
Transmissivity	REAL6	12	0.0..1.0	Ratio of energy transmitted by an object to the amount of energy incident upon it
Two-D Coordinate File Name	STR	17	--	The name of the two-dimensional coordinate file included with SIF/HDI culture data
UL Corner X/Y Image Coordinates (GDS)	INT2D	15	(-999999.. 999999, -999999.. 999999	X/Y cartesian coordinates of the upper left corner of the image

Field Name	Type	Length (CHARS)	Range	Description
UR Corner X/Y Image Coordinates (GDS)	INT2D	15	(-999999.. 999999, -999999.. 999999	X/Y cartesian coordinates of the upper right corner of the image
User Defined FACS Table File Name	STR	17	--	The name of the user defined FACS table file included with SIF/HDI models or SIF/HDI culture data
User Defined Header Data Length (GDS)	INT	5	00000..99999	The length in bytes of data defined by the user to be used for information not currently defined in the NITF header
User Defined Image Data Length (GDS)	INT	5	00000..99999	The length in bytes of data defined by the user to be used for information not currently defined in the NITF image sub-header
User Defined Terrain Data Length (GDS)	INT	5	00000..99999	The length in bytes of data defined by the user to be used for information not currently defined in the NITF sub-header
Vertex List Position	INT	10	0..2147483647	The pointer to a vertex in a vertex table of a model
Vertex Pointer	INT	10	0..2147483647	The pointer to a vertex in a vertex table of a culture tile
Vertex Table File Name	STR	17	--	The name of the vertex table file included with SIF/HDI models
Vertex to Vertex Texture Reference Table File Name	STR	17	--	The name of the vertex to vertex based texture reference table file included with SIF/HDI models
Vertical Captured Texel Size (GDS)	REAL10	16	0.0.. 1.393796575e+42	Approximate ground distance for a texel (expressed in meters) in the vertical y-direction

Field Name	Type	Length (CHARS)	Range	Description
Vertical Resolution (GDS)	REAL6	12	0.0..1.93428e+25	Vertical length of a texel in meters, e.g., 1.0 M/texel
Vertical Size (GDS)	REAL6	12	0.0..1.93428e+25	The vertical size of the entire image in meters, e.g., 1000.0 meters
Visible Range	INT	10	0..2147483647	Distance that a light feature can be seen, expressed in meters
West Tile Neighbor ID (GDS)	INT	10	0..2147483647	The identifier of the neighboring areal specific image to the west; used only for Stage 3 areal textures
Width	REAL10	16	0.0.. 1.393796575e+42	Width of an object, expressed in meters
Wrap	(BOOLEAN; BOOLEAN; BOOLEAN; BOOLEAN)	7	(T, F; T, F; T, F; T, F)	Flag indicating whether a texture pattern can be wrapped along its left, right, top and bottom edges while maintaining a "seamless" appearance
Wrap Type	ENUM	6	NO, NRM, MIRROR	Flag indicating type of texture wrapping performed (for NO, no wrapping is performed; for NRM, right edge aligned with left edge or top edge aligned with bottom edge; for MIRROR, each texture instance is mirrored from the previous texture instance)

APPENDIX B

This page intentionally left blank.

SIF/HDI FACS CODES AND SIF SPECIFIC FEATURE DESCRIPTOR CODES

10. SCOPE

10.1 Scope. This Appendix is a mandatory part of the standard. The information contained herein is intended for compliance.

10.2 Purpose. The purpose of this Appendix is to define the SIF description of the SIF/HDI FACS Codes and SIF specific Feature Descriptor Codes (FDCs) that may be used during the transmission of SIF databases.

20. APPLICABLE DOCUMENTS

20.1 Government documents.

20.1.1 Specifications, standards, and handbooks. The following specifications, standards, and handbooks form a part of this Appendix to the extent specified herein. Unless otherwise specified, the issues of these documents shall be those listed in the issue of the Department of Defense Index of Specifications and Standards (DoDISS) and supplement hereto, cited in the solicition (see 6.2 of this Standard).

MIL-STD-1820 Generic Transformed Data Base Design Standard

(Unless otherwise indicated, copies of federal and military specifications, standards, and handbooks are available from the Naval Publications and Forms Center, ATTN: NPODS, 5801 Tabor Avenue, Philadelphia PA 19120-5099.)

20.2 Order of precedence. In the event of a conflict between the text of this Appendix and the references cited herein, the text of this Appendix shall take precedence. Nothing in this Appendix, however, supersedes applicable laws and regulations unless a specific exemption has been obtained.

30 DEFINITIONS AND ACRONYMS

30.1 Definitions. The definitions provided in this Standard shall apply to this Appendix.

40 GENERAL REQUIREMENTS

40.1 This Appendix shall be a mandatory part of the standard. The information contained herein is intended for compliance.

40.1 FACS Commonality. The starting point for all FDCs and FACS codes within the SIF Military Standard shall be the Glossary of Feature/Attribute Definitions as published by the Defense Mapping Agency (DMA).

40.2 GTDB Commonality. The Feature Descriptor Codes (FDCs) defined in Appendix B of MIL-STD-1820 shall be used within SIF data sets. Additional SIF-specific FDCs shall conform to Section 50 of this Appendix.

50 DETAILED REQUIREMENTS

50.1 FACS Codes. The specified FACS code shall be used for the entries in the FACS Table supplied with a model or a culture tile. Valid ranges and types (integer, floating point, etc.) shall be as defined in the Project 2851 Data Base Design Documents and appendices. The exceptions to this are noted with an asterisk (*), and the acceptable ranges shall be as defined in the DMA FACS Glossary for these codes. To maintain compatibility with the DMA FACS Glossary, the SIF attributes that are represented via the DMA FACS Glossary codes shall use the same three character attribute identifier as used by the DMA FACS Glossary. When a SIF user creates a user-defined FACS attribute, the first two characters of the FACS code shall be the originator code for that user.

SIF Attribute Name	FACS Code	
Absorptivity	ABSORP	
Accuracy Category	ACCxxx	*
Aircraft Facility Type	AFTxxx	*
Amusement Park Structure	APSxxx	*
Angle of Orientation	AOO	*
Angle of Orientation, Derived	1AO	*
Angle of Radar Reflector, Derived	1AR	*
Area Coverage Attribute	ARA	*
ATS Use Attribute	AUAxxx	*
Bank Height Left 1	HL1xxx	*
Bank Height Right 1	HR1xxx	*
Base Polygon ID	BASEID	
Beacon Type Category	BEAxxx	*
Bottom Material Composition	BMCxxx	*
Bridge and/or Superstructure Category	BSCxxx	*
Bridge Reference Number	BRN	*
Brush/Undergrowth Density Code	BUDxxx	*
Building Function Category	BFCxxx	*
Bypass Condition Category	BCCxxx	*
Centroid (Deleted)	CENTRD	
Cluster ID	CLUSTR	
Color Table Index	CTINDX	
Component Name	CMNAME	
Conspicuous Object Category	COCxxx	*
Crane Attribute	CRAxxx	*
Culture Centroid	CCNTRD	
Current Type Category	CURxxx	*
Cycle Rate Off Time	LTOFF	
Cycle Rate On Time	LTON	
Dense Bank Vegetation Left	DVLxxx	*
Dense Bank Vegetation Right	DVRxxx	*
Density of Roof Cover, Derived	1DRxxx	*
Density of Structures, Derived	1DSxxx	*
Density of Tree Cover, Derived	1DTxxx	*
Density Measure, % of Roof Cover	DMRxxx	*
Density Measure, % of Tree/Canopy Cover	DMTxxx	*
Density Measure, Structure Count	DMS	*
Depth, Derived	1DE	*
Depth Below Surface	DEP	*
Depth of Water	DW1xxx	*
Diffuse Reflectance	DISREF	
Direction of Flow	DOF	*

SIF Attribute Name	FACS Code	
Directionality	LTDIRN	
Directivity (Infrared)	IDIREC	
Directivity (Radar)	RDIREC	
Directivity	DIRxxx	*
Distance from Shore	DFS	*
Elevation Point Significance	EPS	*
Embedded Obstruction Code	EOCxxx	*
Emissivity	EMSVTY	
Existence Category	EXSxxx	*
Exitance	EXTNCE	
Exposed Portion Attribute	EPAxxx	*
Farming Type Category	FTCxxx	*
Feature Descriptor Code	FDC	
Feature Identification Code	FID	*
Feature Onset	FTRONS	
Fixed Order Priority	FOPRI	
Gap Width (Measured)	GWM	*
General Roughness Category 1	GR1xxx	*
General Roughness Category 2	GR2xxx	*
General Roughness Category 3	GR3xxx	*
General Roughness Category 4	GR4xxx	*
General Roughness Category 5	GR5xxx	*
Greatest Horizontal Extent	GHE	*
Height	HGT	*
Height, Derived	1HT	*
Height of Areal Feature	2HT	*
Hydrographic Category	HYCxxx	*
Hydrographic Depth	HDP	*
Hydrographic Form Category	HFCxxx	*
Hydrographic Location Category	HLC	*
Hydrographic Origin Category	HOCxxx	*
Hydrographic Seasonal Attribute	HSAxxx	*
Hypsography Portrayal Category	HQCxxx	*
Identification Number	IDN	*
Internal Material Category	IMC	*
Internal Material Volume	IMV	*
Landmark Category	LMCxxx	*
Lane/Track Characteristics	LTCxxx	*
Lane/Track Number	LTN	*
Layer Number (Infrared)	IRLAYR	
Layer Number (Radar)	RLAYER	
Layer Number (Visual)	LAYER	
Length, Derived	1LN	*
Length/Diameter	LEN	*
Length of Cab	LEC	*
Length of Cab (Crane), Derived	1LC	*
Length with Greater Precision	LGP	*
Light Characteristic Category	CHAxxx	*
Light Function Attribute	LFAxxx	*
Light Horizontal Center	LTHCEN	
Light Horizontal Fall	LTHFAL	
Light Horizontal Width	LTHWID	
Light Intensity	LTINTY	
Light Type	LTTYPE	
Light Vertical Center	LTVCEN	
Light Vertical Fall	LTVFAL	

SIF Attribute Name	FACS Code	
Light Vertical Width	LTVWID	
Light Visibility Range	LVR	*
Load Class Type 1	LC1	*
Load Class Type 2	LC2	*
Load Class Type 3	LC3	*
Load Class Type 4	LC4	*
Location/Origin Category	LOCxxx	*
Long Lineal	LNGLIN	
Low Level Effects	LLVEFF	
Material Class Category	MCCxxx	*
Material Composition Primary	MCPxxx	*
Material Composition Secondary	MCSxxx	*
Maximum Edges Per Polygon	MAXEDG	
Maximum Height	MAXHGT	
Median Category	MED	*
Mining Category	MINxxx	*
Missile Site Attribute	MSAxxx	*
Missile Site Type	MSTxxx	*
Mode of Transport	MOTxxx	*
Model Centroid	MCNTRD	
Monitor Type	MNTRTY	
Name Category	NAM	*
Number of Spans	NOS	*
Number of Structures	NUMSTR	
Object Volume	OBJVOL	
Overhead Clearance Category	OHC	*
Overlay Category	OVCxxx	*
Percent of Roof Coverage	DMRxxx	*
Percent of Tree Coverage	DMTxxx	*
Placement Point	PLACPT	
Polygon Illumination Type	ILLUMN	
Polygon Landing Light Illumination	LANDLT	
Polygon Non-Occulting	NONOCC	
Polygon Non-Shadow	NONSHD	
Polygon Normal	NORMAL	
Power Plant Category	PPCxxx	*
Predominant Height	PHT	*
Predominant Height, Derived	1PH	*
Product Category	PROxxx	*
Radar Significance Factor	RSFxxx	*
Radio Direction Finding	RDFxxx	*
Radio Navigation/Communication	NSTxxx	*
Radius	RADIUS	
Railroad Attributes	RRA	*
Railroad Power Source	RPSxxx	*
Railroad/Road Categories	RRCxxx	*
Railroad Gauge Category	RGCxxx	*
Rail Siding Attribute	RSAxxx	*
Religious Denomination	RELxxx	*
Reflectance	RFLNTC	
Road Interchange Type	RITxxx	*
Rock Formation Type	RKF	*
Rock Strata Type	RKSxxx	*
Road Surface Type	RSTxxx	*
Roof Type	SSRxxx	*

SIF Attribute Name	FACS Code	
Sand Dune Orientation	SDO	*
Secondary Material Characteristics	CSMxxx	*
Self-Emitter	SLFMTR	
Shading Range	SHDRAN	
Shading Type	SHADNG	
Shape Code	SHAPCD	
Shoreline Type Category	SLTxxx	*
Slope/Gradient Category	SGC	*
Slope Gradient Left 1	SL1xxx	*
Slope Gradient Right 1	SR1xxx	*
Slope Polygon Range	SPRxxx	*
Soil Type Category	STCxxx	*
Specular	SPECLR	
Spring/Water-Hole Type	SWT	*
Spring/Well Characteristics	SCCxxx	*
State of the Ground	SOGxxx	*
Stem Diameter Size Range 1	SD1xxx	*
Structure Shape Category	SSCxxx	*
Structure Shape of Roof	SSRxxx	*
Structure Shape of Tank Top	STTxxx	*
Superfeature ID	SFRID	
Surface Material Category	SMC	*
Surface Material Subtype	SMCSUB	
Surface Roughness Qualifier	SRQxxx	*
Surficial Material Depth Category	SDCxxx	*
Text Attribute	TXT	*
Texture Map Reflectance	TEXRFL	
Tidal/Non-Tidal Category	TIDxxx	*
Translucency	TRANSL	
Transmissivity	TNSMVT	
Transportation Use Category	TUCxxx	*
Tree Category	TRExxx	*
Tree Spacing Range 1	TS1xxx	*
Underbridge Clearance Category	UBC	*
Use Status	USExxx	*
Vegetation Characteristics	VEGxxx	*
Vegetation Type Category	VGCxxx	*
Visible Range	VISRNG	
Volume/Occupancy Level	VOL	*
Water/Salinity Category	WSCxxx	*
Water Velocity Average	WVAxxx	*
Weather Type Category	WTCxxx	*
Width	WID	*
Width, Derived	1WD	*
Width of Interchange, Derived	3WD	*
Width with Greater Precision	WGP	*
Z Value	ZVL	*

50.2 SIF-Specific FDCs. The Feature Descriptor Codes (FDCs) identified in Appendix B of the GTDB Military Standard (MIL-STD-1820) shall be used for SIF. In addition, the following SIF-specific codes shall be used.

SIF FDC Code	Description
GRV02	Anti-Aircraft Artillery Vehicle
GRV01	Truck, General Purpose
HEL01	Helicopter
SML01	SAM Launcher
SMS01	SAM Site

60 NOTES

(This section contains information of a general or explanatory nature that may be helpful, but is not mandatory.)

6.1 Referenced documents. The following documents were used a references, in preparation of this Appendix.

AMERICAN NATIONAL STANDARDS INSTITUTE

ANSI X3.27 — Information Systems - File Structure and Labeling of Magnetic Tapes for Information Interchange

ANSI/IEEE STD 754 — Binary Floating Point Arithmetic

(Application for copies should be addressed to American National Standards Institute, 11 West 42nd Street, New York NY 10036.)

DEFENSE INTELLIGENCE AGENCY

DDM-2600-63220-90 — National Imagery Transmission Format (NITF), Version 1.1, 1 March 1989, sections 1 through 4.5

(Application for copies should be addressed to Defense Intelligence Agency, DIA/DM-1A, 3100 Clarendon Boulevard, Arlington, VA 22201-5317.)

DEFENSE MAPPING AGENCY

DMA Standard Supporting Mark 90, Section 100, Glossary of Feature/Attribute Definitions, Second Edition, June 1988, revised December 1988.

(Application for copies should be adressed to Defense Mapping Agency, 8613 Lee Highway, Fairfax VA 22031-2137)

DIGITAL EQUIPMENT CORPORATION

AA-LA06A-TE Guide to VMS Files and Devices, Appendix B, "VMS ANSI-Labeled Magnetic Tape," April 1988.

VMS Backup Utility Manual, April 1988

(Application for copies should be addressed to Digital Equipment Corporation, P.O. Box CS2008, Nashua NH 03061)

U.S. DEPARTMENT OF COMMERCE

Initial Graphics Exchange Specification (IGES), Version 4.0, June 1988, sections applicable to CSG

(Application for copies should be adressed to U.S. Department of Commerce, National Bureau of Standards.)

INTERACTIVE COMPUTER MODELLING, INCORPORATED

General Information Manual, May 1988.

(Application for copies should be addressed to Interactive Computer Modelling, Inc, 12200 Sunrise Valley Drive, Suite 210, Reston VA 22091.)

PLANNING RESEARCH CORPORATION

PRC-2851-DBDD-3 Data Base Design Document (DBDD), Standard Simulator Data Base (SSDB), Project 2851 (F33657-86-C-0182)

PRC-2851-DBDD-5 Data Base Design Document (DBDD), Appendix I, Data Type Dictionaries for Project 2851 (F33657-86-C-0182)

(Application for copies should be addressed to PRC, 1500 Planning Research Drive, McLean VA 22102.)

(Non-Government standards and other publications are normally available from the organizations that prepare or distribute the documents. These documents also may be available in or through libraries or other informational services.)

This page intentionally left blank.

APPENDIX C

This page intentionally left blank.

RATIONALE AND GUIDANCE

10. SCOPE

10.1 Scope. This appendix is not a mandatory part of this standard. The information contained herein is intended for guidance only.

10.2 Applicability. This appendix serves to provide supplementary information which may be helpful to users of the SIF standard. For ease of use, it has been organized in a format paralleling that of the standard. In particular, Section 50 of this appendix has been designed to correlate exactly with Section 5 of this standard, to allow users to easily move between them, while removing unnecessary descriptive information from the standard itself.

10.3 Application Guidance. The following paragraphs discuss the manner in which the SIF standard is intended to be applied by acquisition programs. This implementation approach is consistent with the operational methods of the DOD Simulator Data Base Facility (SDBF), which was established expressly to facilitate the exchange and maintenance of training simulator data bases. Deviation from the guidance stated herein may have the undesirable effect of rendering SIF-compliant data sets incompatible with the SDBF, thereby minimizing their accessibility, and hence value, to subsequent programs.

10.3.1 Intended Uses of SIF. Any application of SIF essentially constitutes an interface between two or more data base generation systems (DBGSs), one of which is always expected to be that of the SDBF. Since the SDBF is expected to play a central role in the interchange of SIF data sets, those operations which are performed by the SDBF will be referred to as internal, while corresponding activities performed using other DBGSs will be called external. Any training simulator program may use the SIF standard to define a source of information, a product, or both. External programs which use SIF as a source will be hereinafter referred to as consumer programs; those which deliver SIF as a product will be called producer programs. This section discusses issues associated with the application of the SIF standard data base, from both the producer and the consumer perspective.

10.3.1.1 Producer-to-SDBF. SIF was originally designed as an exchange format to allow externally-generated data bases to be used to populate the SSDB. Accordingly, SIF defines the output product of an external producer's DBGS, and correspondingly defines a source input format for the internal DBGS of the SDBF. Training simulator contracts having significant real-time data base production requirements should always include the requirement that these data bases be delivered to the SDBF in SIF format as well, to facilitate their integration into the SSDB for reuse by later programs. Simply tasking the contractor to deliver data bases in compliance with this standard is not sufficient to guarantee useful SIF products, however; ongoing dialog with a representative of the SDBF is essential, inasmuch as the value of a particular SIF data set must always be assessed relative to other SSDB holdings.

10.3.1.1.1 Frequency of Application. Because it is anticipated that the SDBF will not have adequate resources to do a significant amount of SSDB production internally, it is likely that external producers will contribute the bulk of the SSDB content, at least until the SSDB is amply populated. This being the case, it is envisioned that SIF will frequently be applied as a delivery requirement in the short term. Over time, as the SSDB holdings increase, the need for external production is likely to diminish, and SIF utilization in this manner will become less frequent.

10.3.1.1.2 Application Options. When applied as a requirement upon an external producer, the SIF standard may be used to pass one of two data bases to the SDBF: either the DBGS's embedded source data base, or a real-time simulation data base generated from it. These two data sets may be significantly different; the embedded data base is usually filtered and transformed to create a "flyable" real-time data base. SIF imposes no specific restrictions against the use of one or the other, but it is generally preferred that, when the option exists, the embedded source data base be used as the basis for SIF exchange.

10.3.1.2 SDBF-to-Consumer. SIF is supported as a source of data for external DBGSs, through its implementation as one of the two output products of the SDBF. Accordingly, the SIF standard needs to be cited in contracts desiring to use SDBF products in this form. SIF products furnished to consumer programs are to be obtained through the SDBF, rather than from external producers, in order to ensure that they correctly reflect the merged content of the SSDB, and are properly certified as compliant with the SIF standard.

10.3.1.2.1 Frequency of Application. Inasmuch as the use of SIF as a source implies the need for a copy of the full SSDB, it is not expected that SIF will be widely used for this purpose. The other SDBF product, GTDB, is much better suited to applications for which only a subset of the SSDB is required, as GTDB generation allows for the selective filtration of the data in numerous ways. It is likely that SIF will be used only by consumers who possess their own DBGSs, and have the software tools necessary to process great quantities of information efficiently. It is recommended that individual programs evaluate the use of SIF versus GTDB based upon their specific data base requirements, and the ease with which each can be accommodated by their respective contractors.

10.3.1.2.2 Application Options. When furnishing a SIF data set to a consumer program contractor, the Government may require that it be augmented by the contractor using other source materials, or may require that it be used without further modification. In the first case, there is an implicit assumption that the SIF data is in some way incapable of meeting all requirements of the consumer program, and that additional contractor effort will need to be applied in order to meet them. In the second case, the assumption is that the SIF data is fully capable of meeting all consumer simulator requirements. In either case, it is important that the content of the SSDB in the area of interest be evaluated with respect to the specific requirements of the consumer system, prior to contractual implementation.

10.3.1.3 Binary Application. In some cases, it is desirable to invoke the SIF standard as both an input and output requirement, treating the external program as both a producer and a consumer. One instance in which this may be done is the first case cited under 10.3.1.2.2 above, wherein a consumer program may be required to augment a Government-furnished SIF data set. Assuming that this augmentation is necessary to overcome some inherent deficiency with the SIF data set (and, by association, the SSDB), it will likely be desirable for the SDBF to obtain the enhanced version, such that the SSDB can be subsequently populated with the "improved" information.

10.3.1.4 Producer-Consumer Interaction. It is important that both the SDBF and the external producer or consumer of a SIF data base have a common understanding of the specific requirements of the data base interchange between them. SIF is not a "hands-off" standard; compliance with it cannot simply be written into a contract, and expected to achieve good results, without a mutual understanding of its implications in the context of the specific application. There are numerous variables associated with any particular data base, and it it must be recognized that SIF data sets will exhibit some variability across producers. The standard makes an effort to quantify this variability to the greatest extent practical, but the most effective, efficient use of the SIF as a data base exchange medium can only be realized through an ongoing dialog between the SDBF and its external producers and consumers.

10.3.2 Adaptive Format. As a comprehensive simulator data base format, SIF necessarily supports more data items than are likely to be contained in any given data base. It is conceptually a superset of all commonly used data items. As a result, for any single application of the SIF standard, some parts of the data format may be treated as non-applicable, and hence are regarded as options. Optional items are labeled as such throughout the body of this standard.

10.3.2.1 Mandatory Information. The SIF standard identifies those files, records, and fields which must be populated whenever the SIF standard is invoked, and those which may be left as options to be populated only when the data are readily available. In general, items are defined as mandatory if their absence would significantly compromise the usefulness of a SIF data base, as determined by the design and implementation of the SSDB. In any program requiring the external production of SIF-compliant data sets, the omission of mandatory items must be considered a breach of contract, unless a specific exception has been granted by the acquisition agency, as discussed below.

10.3.2.2 Optional Information. Unlike mandatory items, the producer of a SIF data set is not contractually obligated to populate optional fields. Data items labeled as optional within the standard should not be regarded as necessarily being of lower interest or priority than mandatory items. In some cases, they are optional because, pragmatically, it is understood that many existing data base generation systems do not capture or maintain such data, and that their omission is adequately compensated for by the greater value inherent in the remaining mandatory items. As a general rule, data items labeled "optional" should not be omitted offhandedly, but should be included whenever it is practical and economically feasible to do so. The SDBF should be involved in any decision regarding the disposition of optional data items.

10.3.2.3 Exceptions. External producers of SIF data sets should, in general, be required to populate as many data items as they have information to support. In each invocation of the SIF standard, however, practical issues of cost and/or schedule may determine whether or not certain data items are populated. Procuring agencies should always require external SIF producers to justify any decision not to populate optional items. There may also be rare situations in which an external producer desires relief from having to populate certain mandatory portions of the standard, based on non-existence of the data and/or prohibitive cost. This should be discouraged, as granting such relief would make the SDBF SSDB of significantly less value to subsequent simulator programs. In any case, a decision to leave fields unpopulated within a deliverable SIF data set needs to be coordinated with the SDBF, not left to the individual program, so that the "greater good" is always considered; it may be well worth the Government's short-term investment in one specific program, in order to obtain the long-term benefit of greater support for future training systems; conversely, the information void left by an omission might be so detrimental to the value of the remaining data that the Government might receive the greatest benefit by forgoing the conversion to SIF altogether.

10.3.3 Information Representation Rules. In addition to the specification of a data format, an equally important aspect of the SIF standard is its establishment of certain rules for the population of data within that format. The imposition of these production standards is necessary in order to achieve a minimum level of data quality, allowing the SDBF to verify the acceptability of the data set. Such standards are needed, because SIF data sets, which may be developed and provided by many different sources, have to be integrated into a composite data base (the SSDB) of consistent quality. In the SSDB, quality consistency is of paramount importance, given the potentially broad dissemination of its contents. In deference to the non-redundancy objectives tendered in the establishment of the SDBF, it is believed that the SSDB must maintain data which is of sufficient quality as to preclude the need for its users to correct deficiencies repetitively.

10.3.4 Data Quality. Inasmuch as the SDBF, due to resource limitations, will likely be unable to perform much data evaluation and correction organically, it is incumbent upon the SIF producers to meet the minimum quality levels as a condition of SIF acceptance. It is left to the sponsors of the individual SIF producer programs to ensure that these quality standards are, in fact, met.

10.3.4.1 Quality Enforcement. There is a distinct difference between internally-produced and externally-produced SIF data sets, in the sense of the SDBF's inability to directly control the information content of the latter. Since a SIF data base produced by the SDBF is extracted directly from the SSDB, it will, by definition, meet all SDBF data quality standards; data falling short of these standards would never have been included in the SSDB initially, and thus will not cascade into the SDBF's SIF products. This cannot be said of externally-produced SIF data sets, for which compliance with the SSDB's internal quality standards cannot be assumed. In order to overcome this uncertainty, a series of quality conformance tests, as defined in section 4.4 of this standard, must be performed on any externally-produced SIF data set, as a condition of the Government's acceptance of the product. Only when these tests have been successfully passed, is the SIF data set eligible for further dissemination, as well as inclusion in the SSDB.

10.4 Tailoring. As a comprehensive simulator database format, SIF necessarily supports many more data items than is likely to be contained in any given database. It is conceptually a superset of all commonly used data items. As a result, for any single application of the SIF standard, some parts of the data format may be treated as not applicable.

10.4.1 SIF designers have attempted to identify those portions of the standard which must be populated whenever the SIF standard is invoked, and those which may be left as options to be populated only when the data are readily available. Optional items are labeled as such throughout the body of this Standard. In general, items are defined as mandatory if their absence would significantly compromise the usefulness of a SIF database.

10.4.2 In addition to specifying data formats, the SIF standard includes certain rules for populating the data within the formats. For example, the SIF culture data format specifies that an areal feature shall have its vertices listed in counter-clockwise sequence as viewed from above. Many simulator systems follow this convention, but there are also some systems which have adopted the opposite (clockwise) convention. In such cases, SIF designers have established a common approach, not because the alternatives are "wrong," but simply to make it possible for systems to share data without confusion. Wherever specific conventions are defined in the SIF standard, compliance by SIF producers is mandatory.

10.4.3 The benefit of including production conventions within the standard is that SIF consumers only have to be able to process data in conformance with the selected convention. Allowing greater flexibility (such as a lack of conventions) would reduce the amount of conversion required by SIF producers, but SIF consumers would have to be able to accommodate many different production techniques. Since the number of SIF consumers is expected to exceed the number of producers (due to the fact that every SDBF customer system is an indirect SIF consumer), the establishment of standard conventions was determined to be the better alternative.

10.4.4 In most cases where a convention was deemed necessary, SIF designers have settled on a single approach. However, in some cases the SIF standard supports two or more alternative conventions, giving the exporter a choice. For example, the SIF gridded data format supports multi-band imagery structured in either band interleaved or band sequential formats. In this case, the producer may select whichever alternative is more convenient; the user must be prepared to handle either of the alternatives.

10.4.5 In several sections of the SIF standard, there is provision for use of user-defined data fields. This feature is intended only to support exchange of system-specific data items not explicitly supported by the SIF. It allows the standard adaptable to rapid change, as new technologies require information to be added to simulator databases.

10.4.6 From the external producer's standpoint, the availability of user-defined fields makes SIF flexible enough to be able to capture any essential database elements not anticipated by SIF designers. For instance, if a producer has collected or generated some previously uncollected feature characteristics useful for simulating a new type of sensor, then this data need not be "thrown away" just because the existing SIF standard did not set aside explicit data fields for the new characteristics. In this scenario, the exporter would be expected to define new user-defined FACS attribute records within the SIF feature data files. The meaning of the new records would be defined in the User-Defined FACS dictionary records, which could be stored in the SDBF SSDB for future reference.

10.4.7 From the configuration control perspective of the SDBF; however, this flexibility may be viewed as a drawback, in that each SIF database may contain unique non-standardized data elements. An additional danger with the availability of user-defined fields is that uncontrolled use of this feature is likely to result in abuse. It may be used as a way to avoid the trouble of performing conversions from internal formats to the published SIF standard. As a simplistic example, a system storing object dimensions in feet rather than meters may be tempted to simply write out English-unit dimensions as user-defined FACS values, rather than go to the effort of writing software to convert the dimensions to metric units. Another system may express dimensions in yards, and do the same thing. Over a period of time, there may be numerous such liberties taken with the user-defined field capability, resulting in a data base with dimensions specified in many different units. This would defeat the purpose of standardization, and as such, needs to be avoided. Procuring agencies need to require SIF producers to justify each use of user-defined attributes as unavoidable, because there is no equivalent field in SIF, or because the cost of converting to the nearest SIF equivalent would be prohibitive. On the other hand, whenever user-defined attributes can be legitimately justified, their use should be encouraged, since the only alternative would be to leave the data out of the SIF database.

10.4.8 It is essential that any new use of user-defined fields within SIF databases be coordinated with the SDBF. As the central hub of the SIF user community, the SDBF will be in a position to assign unique field identifiers, maintain a master dictionary of user-defined fields, and reconcile conflicting usages. As time goes on, data items originally introduced into SIF databases as user-defined items may be incorporated into the SIF standard explicitly. It will be the responsibility of the SDBF to make the requisite changes to the standard, in such cases.

10.5 Method of Reference. (Self-Explanatory.)

20. APPLICABLE DOCUMENTS

20.1 The documents called out in section 2 of this Standard apply to this appendix.

30 DEFINITIONS AND ACRONYMS

30.1 Definitions. The definitions provided in this Standard shall apply to this Appendix.

40 GENERAL REQUIREMENTS

40.1 External system interface. This section is self-explanatory in the standard.

40.2 Physical medium. This section is self-explanatory in the standard.

40.3 Quality assurance. Simulator databases are built to widely varying quality (accuracy and reliability) standards from widely varying data sources. In order to support their distribution to other users, it is necessary to establish a common level of quality, which must be met by all producers.

a. Critical information which must be supplied by exporters of SIF databases are quality descriptors. Since the mere existence of the SIF format does not imply any specific quality standards, the SIF standard must also define quality characteristics. To fail to define this information would leave importers of the database completely in the dark as to the database's applicability and reliability for their particular applications.

b. In general terms, the quality of a database may be judged by knowing what data sources were used, what compilation criteria were used to extract data from the sources, and what accuracy standards (if any) were enforced in the automated and/or manual processing of the data. In describing a data source, the producer needs to identify the source product, its currency (date of compilation), and the producing agency.

c. SIF quality descriptors include textual fields for describing the data sources used, and for listing compilation criteria. SIF also includes fields for defining numerical accuracy standards (e.g., circular positioning error) when applicable. The SIF formats allow for specification of multiple sources for a given data unit. SIF also supports different levels of granularity of sources. For example, it is possible to tag a culture file with the source(s) used to compile it; at the same time, it would be possible to tag each individual feature within that file as having been extracted from a unique source; moreover, it would even be possible to tag each FACS attribute of each feature with its own unique source. The finer the detail, the better the SDBF's ability to make quality evaluations.

d. Users of SIF databases should keep in mind that there are multiple accuracy dimensions in a simulator database. Geodetic positioning error, which is often cited as a key quality criterion, is just one measure of accuracy. Also important are the accuracy of the dimensions of objects, as well as feature attributes such as colors and surface materials. When dealing with a processed database, it is important to understand what is real, what is synthetic, and what got left out. The currency of the data is important, in that data that was captured to very high accuracy standards may no longer be an accurate reflection of reality, due to the passage of time. Data captured accurately may also be periodically inaccurate due to temporal variations. Evaluating the accuracy, or overall quality, of a database is thus a complex task. It is for this reason that SIF demands that certain criteria be met allowing the SDBF to make an informed judgment on the overall quality and applicability of a database.

e. In the case of pre-existing databases being converted to SIF, the original data sources may not be known. In such cases, a waiver to the SIF quality standards may be considered. At a minimum, however, the SIF producer must document what is known, and what is unknown. Sometimes, data quality may be implied from the application system so the producer needs to identify the application system as the source.

f. SIF establishes certain minimum quality standards to be observed by future database producers when required to be SIF-compliant.

g. Since DMA will continue to be the default source for validated DoD terrain and culture data, producers of DoD simulator databases will be required to compile terrain and culture data to the same minimum standards applied to DMA standard products. Within the SSDB, DMA Digital Terrain Elevation Data (DTED) is the default terrain source, and Digital Feature Analysis Data (DFAD) serves as the default culture source.

h. DMA specifications contain certain accuracy standards for both DTED and DFAD products. For example, the positioning accuracy of DFAD culture vertices must be within 130 meters, circular error, at the 90% confidence level, relative to the World Geodetic System (WGS) datum. The elevation values in a DTED manuscript are required to be accurate within plus or minus 30 meters, linear error, at the 90% confidence level, relative to mean sea level (MSL) as a datum. These standards will be used as the default values for SIF data sets.

i. For geo-specific photo texture, positioning accuracy of the pixels must meet the same standard as applied to DTED terrain posts, i.e., 130 meters, circular error, at the 90% confidence level, relative to WGS.

40.3.1 General Approach. SIF compliance will be assured in two ways: through the certification of the producer's data base generation system, and through the actual testing of individual data sets. Data set testing is expected to be used only during the initial certification of the process, and as occasional "spot checks" to ensure that the DBGS remains compliant during its production lifecycle. Particularly critical SIF data sets, such as those used for mission rehearsal applications, may be explicitly tested, also.

40.3.2 Process Certification. It is expected that many external SIF producers will, over the periods of performance of their individual contracts, end up generating a relatively large number of SIF data sets. Ostensibly, the quality review and approval of each of these could become a major effort on the part of the SDBF. Since the SDBF's resources will be quite limited, this is not seen as a viable approach. Therefore, instead of testing each individual data set, the producing software processes (i.e., the producer DBGSs) will be certified by the SDBF. The SDBF will assign a figure of merit (FOM) to each external data base generation system, certifying it for SIF production at some quality level. The FOM will fall within the range of zero through nine, nine being the highest level of certification attainable, and representing the best quality SIF data sets. The FOM provides a quantitative metric for the three categories of SIF compliance defined in this standard, namely format conformance, source correlation, and SDBF compatibility. The acceptance of a given data set by the SBDF will be based upon this FOM, in conjunction with other factors, as discussed elsewhere in this appendix. Based upon an evaluation of the SIF data sets output by a given DBGS, the system will be assigned a FOM as follows:

0: Format does not conform to all mandatory requirements of the standard

1: Format conforms with all mandatory requirements of the standard, and may support a subset of its optional fields

2: Meets FOM-1, and supports all optional data fields

3: Meets FOM-1, and populates all mandatory fields with information correlated to its source data base

4: Meets FOM-3, and populates the subset of optional fields with source-correlated data

5: Meets FOM-2 and FOM-3, populates optional fields with source-correlated data when available, and remaining optional fields with default values

6: Meets FOM-3, and creates mandatory source data in accordance with SDBF production standards

7: Meets FOM-4 and FOM-6, and populates the subset of optional fields in accordance with SDBF production standards

8: Meets FOM-5 and FOM-7, and generates all fields in accordance with SDBF production standards

9: Meets FOM-8, and is fully compatible with all internal SDBF maintenance and quality control procedures

40.3.2.1 Format Conformance. Certification of format conformance will basically consist of comparing the SIF data format supported by the DBGS under evaluation with a known standard - specifically, the SIF interface of the SDBF. In the case of external consumers, the SDBF will create a test data set, and it will be up to the consumer to prove that they can read it. Conversely, in the case of the external producer, the producer will create the data set, which will have to be readable by the SBDF. Although this is essentially a simple pass/fail test, there is some flexibility, in that the SIF format includes many optional fields, which do not necessarily have to be supported by all producers. When certifying a process using this technique, one must be mindful of these options; a DBGS certified to generate "optionless" SIF can no longer be considered certified, should it begin outputting SIF data sets which include options. In this case, a recertification at the higher level will be necessary.

40.3.2.2 Source Correlation. This is a more operationally-oriented test than the previous one. In the case of external producers, it ensures that the information content of the SIF data set reflects that of the data base from which it was generated. For external consumers, it guarantees that the information provided by the SDBF actually makes it into the real-time trainer data base. This test a necessary addition to the previous one, since format conformance alone does not guarantee that the correct information is present in the data set.

40.3.2.3 SSDB Compatibility. Of the three stages of SIF certification, this is the most difficult one to quantify, since it deals with production processes which are often subjective or artistic in nature, highly variable even within producers, and usually poorly documented. Basically, the intent of this test is to ensure that the rules by which source material is analyzed, and information is captured from it, are the same for the external producers as within the SBDF. Some of these rules are found in this standard; but many are not, and many will not be fully known until the SDBF has gained operational experience in producing, maintaining, and distributing SIF data sets for different applications.

40.3.3 Data Set Verification (Self-explanatory.)

40.3.3.1 Verification of SIF Product. There are three levels at which compliance with a SIF requirement will be verified. The first level is compliance with the SIF data formats. The second level is the degree to which the SIF output captures the essential elements of the source database. The third level is compliance with SIF data conventions and quality standards.

40.3.3.1.1 Format Conformance. Preliminary verification of compliance with SIF formats is possible via inspection and analysis. Sample records from the output database may be dumped via utility software and compared with the SIF specification. All required data items should be verified as being present, along with relevant optional data items. Individual data items should be inspected for conformance with data dictionary specifications (data type, length, range).

40.3.3.1.1.1 Manual inspection may be augmented with a batch software utility designed to cycle through the entire database searching for deviations from format specifications. The SDBF has a Government-owned format validation utility (written in VAX/VMS Ada) which may be provided to SIF producers as Government Furnished Equipment (GFE).

40.3.3.1.1.2 A more comprehensive verification of compliance would involve a test or demonstration on a system previously verified as capable of accepting SIF inputs. The SIF output database would be input by this second system, verifying that the data are in SIF-compliant format. Again, the SDBF has Government-owned input software which may be shared upon request with systems having compatible hardware/software environments.

40.3.3.1.2 Source Correlation. Once it has been determined that a SIF output conforms with data format requirements, it should be verified that the output captures the essential elements of the source database.

40.3.3.1.2.1 Preliminary verification may be performed by inspection and analysis of the SIF output and comparison with the source database. Sample records from the two databases may be dumped via utility software and compared for functional equivalence. Sampling techniques should be used to verify that all required data items from the source database are present in the SIF output. Utility software may also be used to generate record counts for comparison.

40.3.3.1.2.2 The ideal is a completely lossless conversion from the source database to SIF, such that a duplicate of the original database could be generated from the SIF files. In practice, a certain amount of loss is inevitable; however, this may be acceptable for purposes of database interchange between dissimilar systems. Any such losses should be documented and verified.

40.3.3.1.2.3 A more conclusive, but potentially more expensive, verification approach would be to have the SIF output re-converted to the internal formats of the sending system, so that side-by-side tests and demonstrations between the original and SIF-converted databases may be performed.

40.3.3.1.3 SSDB Compatibility. After a SIF output has been verified as conforming with SIF/HDI data formats and as having successfully captured the source database, it will be verified for compliance with internal SSDB quality standards. This verification step is mandatory inasmuch as any SIF/HDI database must be capable of being integrated into the SSDB.

40.3.3.1.3.1 Preliminary verification of compliance with SIF conventions and quality standards is possible via inspection and analysis. Sample records from the output database may be dumped via utility software and compared with conventions and standards documented in the SIF specification. It is particularly important to verify the accuracy of the information describing to what standards the database was originally built. This includes a list of data sources, compilation criteria, accuracy standards, and processing steps. If such information is not known, then this must be indicated within the appropriate SIF records.

40.3.3.1.3.2 A more thorough verification of compliance with SIF conventions would involve a test or demonstration on a system previously verified as capable of accepting fully-compliant SIF inputs. The SIF output database would be input, processed, and displayed by this second system, verifying that the data are compliant with SIF/HDI standards.

40.3.3.2 Verification of SIF Application. There are two levels at which compliance with a SIF utilization requirement will be verified. The first level is the system's ability to correctly read and interpret the data items received in SIF format. The second level is the degree to which a system can effectively exploit the contents of a SIF database.

40.3.3.2.1 Accommodation. Verification of a system's ability to read a SIF database requires a test database that has previously been verified to be in proper SIF format. The test database may be used to perform a test or demonstration of the system's ability to properly locate, interpret, and store selected SIF data into its internal database(s). Sample records from the internal database may be dumped via utility software and compared with the SIF input data. Utility software may also be used to generate record counts for comparison.

40.3.3.2.1.1 The test should be structured to show that all relevant data items have been extracted from the SIF database, while any non-relevant data items may be ignored. Since the definition of what is and is not relevant is application specific, the test results need to include a complete list of items converted and items ignored, which can subsequently be reviewed.

40.3.3.2.1.2 It is possible that system limitations will cause unavoidable alterations of the SIF data as it is input. For instance, there may be some loss of numeric precision or resolution due to smaller field sizes. Any such conversion losses must be documented for analysis to verify that they are unavoidable and/or acceptable for purposes of the application.

40.3.3.2.2 Utilization. Once it has been verified that a system has properly read and interpreted a SIF input, it is necessary to verify the system's ability to exploit that data in generating a database. The general idea is to verify that relevant SIF data has been incorporated in the system's finished data product.

40.3.3.2.2.1 Verification would occur by demonstrating how SIF data has been incorporated into a deliverable data product. Specific data items contained in the delivered databases should be displayed and traced back to comparable data items in the SIF source. The demonstration may be supplemented by analyses showing that essential characteristics of the SIF data have not been lost or altered by the process, or that any such losses and alterations fall within program specific error tolerances.

40.3.4 Tools and Test Data. (Self-explanatory.)

40.3.5 Test Documentation. (Self-explanatory.)

40.3.6 Exceptions. Inasmuch as the SSDB is the source of all data bases distributed by the SDBF, its quality must be such that it can be applied to the most demanding of training simulator applications, up to and including mission rehearsal. This implies that the SSDB must maintain the highest quality level throughout. It must also be remembered that, once a data set has been incorporated into the SSDB, it is difficult, if not impossible, to remove it, since its integration includes the establishment of connections to other SSDB contents. Under these conditions, it is obvious why it is undesirable to integrate data sets of inferior quality into the SSDB. This imposes the requirement that every SIF data set undergo a demanding quality review as part of its acceptance by the SDBF, with the objective of ensuring that all quality standards are met before considering it for integration. On the other hand, it must be recognized that data base generation is a far cry from an exact science, and that there must be some latitude given when determining what is "acceptable." Standards can be set too high, such that they become an obstacle to the utilization of information which may be quite valuable, in spite of its flaws. For this reason, exceptions will always be seriously considered when a data set fails to meet all of the criteria specified in this standard. Authority for determining whether a given data set meets the SSDB quality standards rests with the SDBF facility manager, as does the ultimate decision of whether to include a data set which in some ways fails to meet these criteria.

40.4 Documentation. The manner in which a SIF data set is documented is of critical importance, from the standpoint of providing sufficient information to the SDBF for evaluation. A hardcopy document allows the facility to make a "quick-look" assessment of the quality and value of the data set, prior to actually expending any resources on processing it through the DBGS. It also allows SIF data sets to be stored on the shelf until they are needed, rather than requiring that they be immediately integrated into the SSDB.

40.4.1 Application. In evaluating a SIF data set for its reuse potential, it is helpful to understand how its source data base was designed to be used originally. Certain applications can impose limitations on data base content, for example, which can render the extracted SIF data unsuitable for different applications. While such limitations will not always preclude the incorporation of the data set into the SSDB, they may serve as indicators of the level of effort which can be anticipated for the upgrade of that information into a more universally applicable form. In rare cases, a deficient data set may be accepted into the SSDB under the assumption that it will never be called upon to fulfill a more stringent requirement than the one for which it was constructed initially. SDBF products generated from this data will carry the appropriate caveats, warning consumers of the unsuitability of the data set for any application other than its original one.

40.4.1.1 Uniqueness. The SIF data set should, above all, describe the unique characteristics of the data set. Although a SIF data set may fall short of the general acceptability criteria, it may be accepted into the SSDB for lack of a superior alternative. Such information will be accepted under the assumption that it will eventually be superseded. Recipients of the data will be informed of its shortcomings, and of the likelihood of its eventual replacement, by the SDBF.

40.4.1.2 Timeliness. In certain instances, a requestor may levy on the SDBF a time-critical requirement for a specific data set, for which coverage does not exist in the SSDB. In order to fulfill this requirement, it may be necessary for the SDBF to obtain a SIF data set which does not meet all of the usual acceptability criteria, and incorporate it into the SSDB. In such cases, the SDBF may immediately distribute the data set to the requestor under the appropriate caveats; alternatively, it may perform work as necessary to bring the data up to a level consistent with the rest of the SSDB, prior to distribution. If the former approach is chosen, the deficient data will not be permanently stored in the SSDB, unless it can be brought up to the facility's internal standards. If it is determined that the data cannot be improved sufficiently to meet these criteria, it will be removed from the SSDB following distribution to the initial requestor.

40.4.2 Training Utility. The intrinsic training value of the candidate SIF data set will be a consideration in the decision regarding whether or not to incorporate it in the SSDB. When a data set contains information which is not source-correlatable, but has been inserted for the purpose of enhancing its merits as a training tool, that data set should not be treated as inferior to a similar one lacking such data. Although there may be many cases in which the training utility of a data set is decisive in allowing its incorporation into the SSDB, it is believed that the specifics of each case are likely to be unique; as such, there can be no "hard and fast" rules governing this determination. These decisions will be made on a case-by-case basis, and require a fair amount of interaction among the external SIF producer, the acquisition agency, and the SDBF.

40.4.3 Content. The SIF data set needs to describe the data set in adequate detail to allow the reviewer to thoroughly understand its content without actually having to inspect the data itself. The level to which this description is carried depends on the nature of the data set itself; a simple DMA-based terrain and feature data base requires far less explanation than one built "from scratch" from a diverse set of maps, photographs, and blueprints, for example. As a minimum, coverage maps should be included, showing the overall gaming area, and locations of high-resolution insets; each model in the the data set should have a corresponding plot, showing its geometry and texturing; there should be tables of statistics showing densities and accuracies on a per-tile basis; individual sources should be cited for each type of information; and so on.

40.4.4 Indigenous Standards. The term "indigenous" refers to the fact that many external SIF producers use their own internal standards when creating models, which may differ from those used by the SDBF. Information included in an indigenous standard can, for example, describe the procedures by which a complex shape is analyzed and dissected into polygons, the placement of separating planes, decisions made based upon image generator performance characteristics, vertex and polygon numbering & sequence scheme, etc.

40.4.5 Transformations. This general heading refers to any processes performed on the data, which may include SDBF-compatible operations, as well as indigenous ones. It is important to know, for example, whether terrain resampling was performed to convert a UTM grid into the geodetic spacing required by the SSDB. It is equally important to know if, once this had been done, an accuracy test was performed comparing the new grid to a DTED cell, for example.

40.4.6 Utilization Instructions. In many data bases, in certain select areas, a few relatively simple features are typically replaced by a large number of elaborate models. An example of this is an airfield, for which the data base developer may substitute a highly detailed model complex for the single lineal runway feature found in the culture data. In such cases, the interrelationships between terrain, culture, texture, and models may become quite complicated, precluding the use of standard automated techniques for their integration into a real-time data base. To support the reuse of this type of data base information, it is necessary for detailed "assembly instructions" to be provided to subsequent users of the data set. At the very least, these instructions need to be documented in the data base descriptive document delivered with the SIF data set. It is also recommended that they be included within the data set itself, in the form of comment records.

50. DETAILED REQUIREMENTS

50.1 Standard Simulator Data Base Interchange Format (SIF)/High Detail Input/Output (HDI) data base. This section defines the overall SIF/HDI data base format, in both a logical form and a physical tape form. The SIF/HDI Data Base Format was designed with the themes of compactness and ease-of-use in mind. When these themes were counter to one another, compromises were made. Data compression is supported in the forms of elimination of leading or trailing blanks within ASCII strings, the use of binary rather than ASCII data for long lists of coordinates, and support for standard image compression techniques. Existing standards were incorporated into the SIF/HDI standard when the existing standard's goals and format were consistent or compatible with those of SIF/HDI. The standards directly incorporated by the SIF/HDI standard include the ANSI/IEEE Standard for Binary Floating Point Arithmetic (ANSI/IEEE Std 754-1985), the ANSI standard for magnetic tape formats (ANSI X3.27-1978), DOD's National Imagery Transmission Format (NITF), and the Initial Graphics Exchange Specification (IGES) produced by the National Institute of Standards and Technology (NIST).

50.1.1 SIF/HDI Data Base Structure. SIF/HDI consists of four general classes of simulator data. These are terrain, culture, models (both CSG and polygonal), and texture. For each application of the SIF/HDI standard, these classes of data shall be included or excluded as appropriate to the sending and receiving applications.

a. Terrain application guidance. The gridded terrain data format is to be used whenever a simulator database represents the conformation of the earth's surface over a gaming area as a matrix (grid) of elevation values. The feature data format should be used instead of (or in addition to) the terrain data format when a simulator database represents terrain using vector graphic primitives (points, lines, areals).

b. Culture application guidance. The culture data format is to be used whenever a simulator database represents features using vector graphics primitives (points, lines, polygons) to describe feature geographic positions and boundaries. Culture data may also be used in addition to terrain data. The photo texture format should be used instead of (or in addition to) the culture data format when a simulator database represents features using raster graphics primitives (pixels, texels).

c. Model application guidance. The CSG and polygonal model formats have been integrated into a single model format for ease of use. The CSG portion of the model format should be used whenever a simulator database developer models specific and generic features using constructive solid geometry primitives (e.g., spheres, cylinders, cubes). The polygonal portion of the model format should be used instead of (or in addition to) the CSG model portion when a simulator database developer models features using polygonal (vector) graphic primitives. The texture format should be used instead of (or in addition to) the model format when a simulator database developer models features using raster graphics primitives.

d. Texture application guidance. The texture format is to be used for photographic, sensor-derived, or generic texture patterns. The texture format should also be used when a simulator database includes features modeled using raster graphics primitives.

50.1.1.1 Logical format. (Self-explanatory.)

50.1.1.1.1 Data base. The SIF/HDI data set is logically divided into four sections, corresponding to the general forms in which the data base information is organized.

50.1.1.1.2 Section. Distinct sections exist for header, model, vector, and gridded data types.

50.1.1.1.3 File/record/field/item. Usually, a field consists of a single item. An example of a field with more than one item is a vertex field where each of the coordinates (X, Y, Z) defining the vertex are items.

50.1.1.2 Physical format. (Self-explanatory.)

50.1.1.2.1 Data order. Each section represents a series of files. Each of the three sections is optional, and their existence is indicated within the SIF/HDI Data Base Header File. For further details on the content of each of these sections as well as the SIF/HDI Data Base Header File, one should consult the appropriate sections of this document.

50.1.1.2.2 Physical tape format. The SIF/HDI tape format is a subset of the ANSI standard. According to this standard, volumes are written and read on 9-track magnetic tape drives only. The standard specifies the format, content, and sequence of volume labels and file labels. All labels must consist of ASCII characters. The ANSI standard specifies a maximum block size of 2048 bytes; however, in accordance with the theme of compactness and with the capabilities of today's commonly available technology, it was decided to allow larger block sizes in SIF/HDI, thus saving large amounts of media for data storage. Larger block sizes tend to be more optimal in tape usage. It is recommended that SIF/HDI data base creators use as large a block size as possible, given the processing capabilities of the systems exchanging data bases. The SDBF system supports up to the maximum block size. The allowable file names in the VMS ANSI implementation used by SIF/HDI are a subset of those in the ANSI standard. Specific file names used by convention are listed in the format descriptions elsewhere in this document. These specific file names are required for compatibility with the SDBF. For more details, one should consult the specified ANSI standard and/or VAX/VMS documentation, including the VAX VMS Magnetic Tape User's Guide.

50.1.1.2.3 General file and data formats. Model and culture data are quite similar in their formats. In general, the SIF/HDI Data Base Header File and all files in the Model Data Section and the Culture Data Section, except where explicitly noted otherwise, are in a compressed ASCII format with record keyword separators and ASCII null ('00') field separators. The Gridded Data Section, containing both terrain and texture data, has its files stored in the specified NITF format. All header files are stored in non-compressed ASCII, while the data files containing the actual grid data are in a binary format as specified by the NITF standard.

50.1.2 SIF/HDI File Formats. (Self-explanatory.)

50.1.2.1 SIF/HDI Data Base Header File Format. This section defines the detailed file, record, and field structure of the SIF/HDI data base header file. This file is used to provide general information on the contents of a SIF/HDI database. The intent of the file is to allow a SIF/HDI user to plan for the data to be input from the data base media. Information, including areas of coverage and file names, are provided for models, culture, terrain, and texture. A compressed form of ASCII has been chosen for the data base header file. Plain ASCII has the advantages of being system-independent, easy to work with, and amenable to visual review. Its main drawback is its voluminousness. Binary has the advantage of compactness, but it is not system-independent at the byte and word level, and it is not amenable to direct visual review.

50.1.2.1.1 Header Data Encoding. Since many records are optional and the number of records of a certain type may vary, a method is needed so that one knows the type of record being read. The SIF/HDI designers considered using either unique keywords for each record type or record counts stored in a preceding record. In the keyword approach, every record begins with a keyword that identifies its type. The advantages to this approach include ease of direct visual review, an easy-to-check built-in quality assurance, and ease of future expandability. Its major disadvantage is that it requires more storage than the record count approach generally would. In the record count approach, a required record would hold counts for record types whose number of occurrences may vary. Counts would be zero for those optional records not used. The main advantage of the count approach is that it would not be as verbose as the keyword approach; however, it has several disadvantages. In visually reviewing records, it would not be as easy to know immediately the type of a record. If a count were incorrect, software recovery would not be as easy as under the keyword approach. Visual verification of data would certainly be easier under the keyword approach. Finally, if new record types would be added in the future, under the count approach, records holding counts would need to be modified, thus invalidating previous SIF data bases and SIF software to read/write such databases. Under the keyword approach, such software would still be valid since it could simply ignore keywords corresponding to new data or data that a particular SIF data base user does not need. Based on these factors, the keyword approach has been adopted in the SIF/HDI. To minimize the impact of additional storage, keywords have been limited to two ASCII characters. While some of the more important record counts have been included in the data base, these are provided primarily for convenience. A need for embedding comment fields or free text fields in the SIF format has been identified.

50.1.2.1.2 Header section structure. (Self-explanatory.)

50.1.2.1.3 Header file structure. (Self-explanatory.)

50.1.2.1.3.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the type of file.

50.1.2.1.3.2 Transmittal Description Record. This mandatory record contains identification information for the entire data base.

50.1.2.1.3.3 Data Directory Record. This mandatory record contains directory information regarding the entire data base.

50.1.2.1.3.4 2D Static Model Library Header File Name Record. This record is mandatory only if 2D static models exist in the data base. The existence of 2D static models is indicated by a count in the Data Directory Record. If they do exist, there shall be exactly one of these files. The record contains the file name for a single 2D static model library header file.

50.1.2.1.3.5 2D Static Model Entry Record. This record is mandatory for each 2D static model in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identification and descriptive information for a single 2D static model.

50.1.2.1.3.6 3D Static Model Library Header File Name Record. This record is mandatory only if 3D static models exist in the data base. The existence of 3D static models is indicated by a count in the Data Directory Record. If they do exist, there shall be exactly one of these files. The record contains the file name for a single 3D static model library header file.

50.1.2.1.3.7 3D Static Model Entry Record. This record is mandatory for each 3D static model in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identification and descriptive information for a single 3D static model.

50.1.2.1.3.8 3D Dynamic Model Library Header File Name Record. This record is mandatory only if 3D dynamic models exist in the data base. The existence of 3D dynamic models is indicated by a count in the Data Directory Record. If they do exist, there shall be exactly one of these files. The record contains the file name for a single 3D dynamic model library header file.

50.1.2.1.3.9 3D Dynamic Model Entry Record. This record is mandatory for each 3D dynamic model in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identification and descriptive information for a single 3D dynamic model.

50.1.2.1.3.10 Model Table File Name Record. This record is mandatory only if models exist in the data base. If models are provided, then there shall be one of these records. The record contains the file names for each of the tables referenced by the models. Each of these files is defined in the section on SIF/HDI models.

50.1.2.1.3.11 Culture Header File Name Record. This record is mandatory only if culture exists in the data base. If culture is provided, then there shall be one of these records. The record contains the file names for each of the files containing culture header information. Each of these files is defined in the section on SIF/HDI culture.

50.1.2.1.3.12 Culture Tile Entry Record. This record is mandatory for each culture tile in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single culture tile. Each of the files is defined in the section on SIF/HDI culture.

50.1.2.1.3.13 NITF Header File Name Record. This record is mandatory only if gridded data exists in the data base. The existence of gridded data is indicated by counts for terrain tiles and all eight types of textures in the Data Directory Record. If they do exist, there shall be exactly one of these files. The record contains the file name for a single NITF header file.

50.1.2.1.3.14 Terrain Tile Entry Record. This record is mandatory for each terrain tile in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single terrain tile.

50.1.2.1.3.15 Generic Texture Entry Record. This record is mandatory for each generic texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single generic texture.

50.1.2.1.3.16 Stage 3 Specific Model Texture Entry Record. This record is mandatory for each stage 3 specific model texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single stage 3 specific model texture.

50.1.2.1.3.17 Stage 2 Specific Model Texture Entry Record. This record is mandatory for each stage 2 specific model texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single stage 2 specific model texture.

50.1.2.1.3.18 Stage 1 Specific Model Texture Entry Record. This record is mandatory for each stage 1 specific model texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single stage 1 specific model texture.

50.1.2.1.3.19 Stage 3 Specific Areal Texture Entry Record. This record is mandatory for each stage 3 specific areal texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single stage 3 specific areal texture.

50.1.2.1.3.20 Stage 2 Specific Areal Texture Entry Record. This record is mandatory for each stage 2 specific areal texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single stage 2 specific areal texture.

50.1.2.1.3.21 Stage 1 Specific Areal Texture Entry Record. This record is mandatory for each stage 1 specific areal texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single stage 1 specific areal texture.

50.1.2.1.3.22 SMC/FDC Texture Entry Record. This record is mandatory for each SMC/FDC texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single SMC/FDC areal texture.

50.1.2.2 Model Data. This section defines the detailed file, record, and field structure of the SIF/HDI CSG and polygonal model data format. The SIF/HDI model format uses the logical formats of the SDBF Standard Simulator Data Base (SSDB) as a starting point. The SSDB stores models in a dual format that includes both Constructive Solid Geometry (CSG) and polygonal geometry definitions. An SSDB model may have only the CSG definition, only the polygonal definition, or both. Other information, such as attributes, are stored only once for the model, regardless of the geometric definition(s) used. The internal CSG format used in the SSDB has some features which are specific to the SDBF software environment. For the purposes of a general exchange format, it was felt that a less system-specific format was desirable. At the present time, there is an industry standard called Initial Graphics Exchange Standard (IGES) used for the industry exchange of CSG models. Rather than re-invent the wheel, SIF/HDI designers decided to base the SIF/HDI format for CSG models on IGES, enhancing it where necessary. The vendor of the commercial software package used to support SDBF modeling (Interactive Computer Modelling Geometric Modelling System (ICMGMS)) plans to support IGES also. Although most image generator vendors use polygonal models, there are sufficient differences in their modeling approaches to make complete model compatibility a difficult objective. Technical differences between models are deeply tied to vendor-specific optimization strategies. However, there is enough inherent compatibility in the basic model geometries and attributes to make exchange of many models possible. The SIF/HDI polygonal model format is intended to be sufficient to pass the geometry of the model along with common model attributes, but it will be left to recipients of these models to address optimization issues. A compressed form of ASCII has been chosen for models. Plain ASCII has the advantages of being system-independent, easy to work with, and amenable to visual review. Its main drawback is its voluminousness. Binary has the advantage of compactness, but it is not system-independent at the byte and word level, and it is not amenable to direct visual review.

50.1.2.2.1 Model Data Encoding. As with the Header Data (para 50.1.2.1.1), a keyword approach has been selected for model data encoding.

50.1.2.2.1.1 Model Building Standards. Models transferred via SIF must follow two basic guidelines to expedite transformation and placement into the SSDB. First, the orientation of the model must follow SDBF standards; otherwise, models could be facing the wrong direction. Second, the origin of the model must be logically defined to facilitate rapid and accurate placement of models onto known coordinates or point features. The front of a model is defined as the primary entrance of a static model. For example, a house model will have the X-axis pointing normal to the polygon that would normally be facing the access road. This polygon will normally, but not necessarily, contain the primary entrance as well as the house number. If the primary entrance is located on the side, the X-axis should still be defined by the polygon facing the street since interactive placement of the model is simplified by making all houses point toward the closest road. A car, truck, plane, helicopter, battleship, cruise missile, space shuttle, or any other moving vehicle should point in the direction it travels. A model cannot be placed in the air unless the origin is translated, to artificially elevate it. An Anti-Aircraft Artillery placement should have its origin along the axis of the turret. A helicopter shall have its origin along the axis of the main rotor. A car shall have its origin in the center.

50.1.2.2.2 Model Section Structure.

a. For CSG models, the internal logical format of the string records is allowed to vary in order to support a wide range of data about a model's geometry and attributes. In the SSDB, a commercial software product (ICMGMS) with a particular implementation of standard CSG commands has been used. In order to serve as a vendor-independent exchange format, the SIF/HDI model format will substitute its internal CSG commands with their IGES equivalents. IGES is the standard mechanism in industry for the exchange of various types of graphical data, including CSG models. IGES Version 4.0 is a U.S. Department of Commerce document dated June 1988 whose distribution is administered by the National Computer Graphics Association (NCGA). Its document number is NBSIR 88-3813. While a version 5.0 has been released, the version 4.0 will be referenced in this document due to its proven maturity.

b. For polygonal models, the geometry of each model is represented as a set of surface polygons in 3-D space. The perimeter of each polygon is described by a set of coordinates or vertices. The polygon is implicitly closed. Each surface or polygon may have descriptive and rendering attributes associated with it. A wide range of fields are available to describe attributes specific to radar, visual, and infrared simulation, as well as general attributes applicable to all three. Each polygon may reference texture maps from an associated Model Texture Library. The model library structure also supports composite models in which one model references another as a component.

c. A valuable feature of the existing SSDB format is its support for a FACS-like attribute coding scheme. Modeled after DMA's Feature Attribute Coding System, the P2851 FACS is a system of self-defining feature attributes, which gives total flexibility to add new attributes to SIF/HDI as the need arises. When exchanging data bases between different systems, this approach is likely to prove highly useful in bridging the gap between different feature and attribute sets. Users are encouraged to take advantage of this approach rather than be constrained by features and attributes explicitly supported in the SSDB at any given point in time.

50.1.2.2.2.1 <u>Field Format</u>. There is much extended use of FACS attributes to avoid the overhead of many fixed attributes which are rarely populated. FACS attribution has already allowed for the definition of a large number of new FACS attributes not currently in the GTDB. Simple table structures have been employed to support flexible FACS attribution, colors in either red-green-blue (RGB) or hue-chroma-value (HCV) formats, and FID/FDC cross-referencing. Flexibility has been incorporated in the mapping of textures to polygons by allowing different methods.

50.1.2.2.2.2 <u>Section Format</u>. (Self-explanatory.)

50.1.2.2.3 <u>Model File Structures</u>. (Self-explanatory.)

50.1.2.2.3.1 <u>Model Library Header File</u>. This mandatory file contains control information describing the library contents.

50.1.2.2.3.1.1 <u>SIF File Identifier Record</u>. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.2.3.1.2 <u>Model Library Header Record</u>. This mandatory record contains control information describing the library contents.

50.1.2.2.3.2 <u>Model Data File</u>. This mandatory file contains identification, description, and control information for a specific model. The file contains information indicating the type of geometric description available for this model: Constructive Solid Geometry (CSG), polygonal geometry, or both. It then contains the model's descriptive information as well as attribute and supporting geometric structure data.

50.1.2.2.3.2.1 <u>SIF File Identifier Record</u>. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.2.3.2.2 <u>Model Header Record</u>. This mandatory record identifies a model, which can consist of one or more actual model geometries representing a given object at varying LODs. If a SIF/HDI data base creator has a model identified with an FDC, then it would be added here as an optional field. If retaining a user-specific FID in the SIF is desired, then a FID code would be supplied here.

50.1.2.2.3.2.2.1 <u>Data Source Table Pointer List Subrecord</u>. This subrecord provides a list of pointers into the Data Source Table.

50.1.2.2.3.2.3 <u>LOD Header Record</u>. This mandatory record describes a particular LOD version of a model.

50.1.2.2.3.2.4 <u>Model Cluster Statistics Record</u>. This optional record gives complexity statistics about one or more polygon clusters within a model LOD. Clusters are separated from other clusters by separation planes. Polygons are grouped into clusters to provide a basis for display-priority resolution when the model is rendered on certain graphics devices. Two-dimensional models in SIF do not contain separation planes and will therefore always constitute a single cluster. Separation planes are optional in three-dimensional SIF models.

50.1.2.2.3.2.5 Separation Plane Record. This optional record is used to define a separation plane within a 3-D model. Separation planes are used to divide a model into distinct clusters of polygons, which provide a basis for efficient display priority resolution when the model is rendered on a graphics device. Under the SDBF system, such clusters are defined to be convex. The P2851 Binary Separation Planes Flag Field in the LOD Header Record shall indicate whether or not SDBF-defined separation planes and cluster IDs are provided.

50.1.2.2.3.2.5.1 The following describes how the separation plane record fields are defined by the P2851 system. If another source for a model uses a different definition for separation plane fields, then that descriptive information should be provided in the SIF via user-defined FACS.

50.1.2.2.3.2.5.2 A Separation Plane Number indicates the position of a separating plane within a binary separating plane (BSP) tree. An example of a BSP tree is illustrated below. At every level of the tree, the left child of a parent tree node represents the "true" (i.e., visible) half-space, or side, of the plane, while the right child represents the false side of the plane.

50.1.2.2.3.2.5.3 In the example, the root node (SP 1) of the BSP tree by itself divides the entire model into two half-spaces or clusters. The root node is shown having a left child and a right child. The left child divides the true cluster of the root node plane into two more clusters. The right child plane does the same to the false cluster of the root node plane. Finally, the right child plane has a left child of its own, dividing that plane's true cluster into two more clusters.

```
        SP 1
       /    \
      /      \
  SP 2      SP 3
             /
            /
        SP 6
```

50.1.2.2.3.2.5.4 As mentioned above, each node, or plane, has an identifying separation plane number which represents its position in the BSP tree. This number is determined by counting from top to bottom within the tree, from the left-most node to the right-most node at each level, as if the tree were complete (i.e., with all levels filled). This explains why the lowest node in the example is numbered 6 and not 4.

50.1.2.2.3.2.5.5 Note that the order of creation of the planes may be partially independent of the position of the planes in the tree, and hence of the separation plane numbers. Of course, the very first separation plane created for a model would have to be the root node and be assigned plane number 1. At lower levels, however, the nodes could be defined in any order, so long as any given node's parent has been previously defined. Within a model, the separation plane records will be physically ordered not by separation plane number but by the order in which the planes were created.

50.1.2.2.3.2.6 Subsidiary Model Reference Record. This optional record is used to designate another model within the model libraries as a subcomponent of this model. The Translation Field is applied to the subsidiary model after the subsidiary model's origin has been placed at the parent model's origin and the two coordinate systems have been aligned. The Scale Factor is applied to the subsidiary model about its own origin. The rotations are done in order about the x-axis, the y-axis, and the z-axis of the subsidiary model's local coordinate system. The FACS Table Index exists in this record in order to support user-defined FACS, particularly articulation parameters.

50.1.2.2.3.2.7 Point Light String Record. This optional record is used to define each point light string within a model. It can be used to represent a single light by indicating that the number of lights is one. Point lights are light emitting objects represented spatially by a single coordinate within a model (e.g., a headlight on an automobile). They contain several attributes necessary for describing a light emitter such as the light lobe parameters, cycle rate, light type, and intensity. Point light strings are a sequence of discrete but logically connected light emitters (e.g., runway lights).

50.1.2.2.3.2.7.1 Point Light Positions Subrecord. This subrecord is used to define the position of each point light within a point light string on a model.

50.1.2.2.3.2.8 Collision Test Point Record. This optional record is used to designate a Vertex record within the Model Vertex File as a collision test point for a model.

50.1.2.2.3.2.9 Model LOD Texture Reference Pointer Record. This optional record is used to point to a texture reference table entry that applies to an entire model LOD. This pointer has the lowest priority of the three types of texture reference pointers (i.e., if a polygon has either a component texture reference pointer or a polygon texture reference pointer, then those pointers will take priority over a model LOD texture reference pointer).

50.1.2.2.3.2.10 IGES Start Record. This record is mandatory only for models with a CSG format. This record indicates the start of the IGES commands.

50.1.2.2.3.2.11 IGES Records. These records are mandatory only for models with a CSG format. IGES records are in the ASCII form as specified by the IGES Version 4.0 Specification. These are 80-byte text records. These records are divided into five distinct sections: Start, Global, Directory Entry, Parameter Data, and Terminate. Each record in a section has a sequence number starting at 1 and then incrementing by 1 for each record within a section. Entities that can be used are limited to the Constructive Solid Geometry Model Entities, curve entities used for solids of revolution or linear extrusion, and transformation matrix entities. The IGES record format uses keywords and ASCII text fields. Refer to the IGES document for more details.

50.1.2.2.3.2.12 Polygonization Instruction Record. This record is mandatory only for models with a CSG format. This record contains parameters used to control the process of conversion of a CSG model to a polygonal representation.

50.1.2.2.3.2.13 Component Header Record. This mandatory record contains a wide variety of descriptive attributes for a component within the model LOD. A component is a part of a model that can be used for graphic manipulation as a single entity and for assignment of common attribute values. For the CSG format model, a component is a collection of primitive shapes. For the polygonal format model, a component is a collection of polygons. A model may be defined to consist of a hierarchy of one or more components, each of which may be made up of one or more sub-components, and so on. The elements making up a component need not be physically contiguous; for example, the four wheels of a vehicle may be grouped as a single component to support one-time attribution of surface material, color, etc.

50.1.2.2.3.2.13.1 The Color Table Index points to a default color value for any of the component's polygons that did not have a polygon color set via a FACS code in the Polygon Header Record. If a model is intended for use only in a non-visual simulation (e.g., radar or infrared) and thus the color is not needed, then a color table shall not exist and the Color Table Index value shall be set to zero.

50.1.2.2.3.2.14 Model Microdescriptor Record. This optional record contains one or more microdescriptors associated with a model component.

50.1.2.2.3.2.15 Component Texture Reference Pointer Record. This optional record is used to point to a texture reference table entry that applies to an entire model component. This pointer has the middle priority of the three types of texture reference pointers (i.e., it has priority over a model LOD texture reference pointer but not over a polygon texture reference pointer).

50.1.2.2.3.2.16 Polygon Header Record. This record is mandatory only if the model has a polygonal format. This record describes the attributes of a polygon within a particular model.

50.1.2.2.3.2.16.1 Each polygon belongs to a group of polygons that form a homogeneous part of the model. This is known as a component. Polygons within a component share many or all of the same attribute values.

50.1.2.2.3.2.16.2 In addition to a Component ID and a unique Polygon ID, each polygon shall have a Cluster ID which associates the polygon with a group of polygons which have been logically separated from the rest of the model for purposes of hidden surface calculations. The meaning of the value of the Cluster ID may vary, depending on the source of the model (found in the Data Source Table). If the source is the SDBF, then the Cluster ID also identifies a polygon's position relative to any of the separation planes defined by the Separation Plane Records associated with the model. The P2851 Binary Separation Planes Flag Field in the LOD Header Record shall indicate whether or not SDBF-defined separation planes and cluster IDs are provided.

50.1.2.2.3.2.16.3 The SDBF-defined Cluster ID value is determined by the polygon's position relative to the separation planes. A polygon can be on the true side of a plane, the false side, or in a "don't care" position. (This assumes that polygons intersecting the plane have already been cut to lie entirely to one side or the other.) The cluster ID is defined as follows: if a polygon lies to the true side of the "nth" plane in the separation plane list, then the "nth" low-order bit is set to '1'; otherwise, it is '0'. The "don't care" case (which arbitrarily takes the value of '0') is one where a polygon has already been separated from the area of concern of a separating plane by a previously placed plane. For further information, see the Separation Plane Record description.

50.1.2.2.3.2.17 Vertex Pointer Record. This record is mandatory only if the model has a polygonal format. This record is used to associate a model polygon with a Vertex record within the Vertex Table File. There will be three or more of these records defining the geometry of each model polygon. By convention, a model polygon will be closed implicitly rather than explicitly; i.e., the first vertex of a polygon will not be explicitly referenced again as the last vertex.

50.1.2.2.3.2.18 Vertex Normal Record. This optional record is used to associate a normal vector with a Vertex record within the Vertex Table File.

50.1.2.2.3.2.19 Vertex Color Record. This optional record is used to associate a color with a Vertex record within the Vertex Table File.

50.1.2.2.3.2.20 Polygon Microdescriptor Record. This optional record contains one or more microdescriptors associated with a model polygon.

50.1.2.2.3.2.21 Polygon Texture Reference Pointer Record. This optional record is used to point to a texture reference table entry that applies to a polygon. This pointer has the highest priority of the three types of texture reference pointers (i.e., it has priority over both a component texture reference pointer and a model LOD texture reference pointer).

50.1.2.2.3.3 Vertex Table File. This binary file is mandatory only for models that have a polygonal format description. Unlike other files in the model section, it is written in binary in order to compress the long list of vertices used for the polygons of a model. The floating point coordinates are written using the ANSI/IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE Std 754-1985.

50.1.2.2.3.3.1 Vertex Record. This mandatory record provides the coordinates of a vertex.

50.1.2.2.3.4 Data Source Table File. This mandatory file contains information on the source(s) used to define a model or an attribute of a model. The source(s) of information used to define each model are documented within one or more Data Source Table records. These sources may be used to make judgments as to the accuracy, currency, and/or reliability of a model. Typically, there will be a single source for all basic data about a model.

50.1.2.2.3.4.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.2.3.4.2 Data Source Table Header Record. This mandatory record contains control information on the contents of the table.

50.1.2.2.3.4.3 Data Source Table Entry Record. This mandatory record contains information on the source used to define a model or an attribute of a model. The source(s) of information used to define each model are documented within one or more Data Source Table Entry records. These sources may be used to make judgments as to the accuracy, currency, and/or reliability of a model. Typically, there will be a single source for all basic data about a model.

50.1.2.2.3.5 FACS Table File. This optional file serves two primary purposes: (1) to minimize space by eliminating redundant model and/or component attribute assignments and (2) to allow expandability of supported attributes. There shall be zero or one of these files in the SIF Model Section. A FACS Table Index pointing to the appropriate table entry can be found in several records.

50.1.2.2.3.5.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.2.3.5.2 FACS Table Header Record. This mandatory record contains control information on the contents of the table.

50.1.2.2.3.5.3 FACS Table Entry Record. This mandatory record in the FACS Table is composed of two control fields and a variable number of FACS Attribute records.

50.1.2.2.3.5.3.1 FACS Attribute Subrecord. This optional subrecord contains a FACS (Feature Attribute Coding Standard) value associated with a model polygon. This record should be used to pass various physical, cultural, or sensor-response characteristics as may be needed to support a simulation. The keywords supported explicitly by SIF/HDI are documented in an appendix to this standard. When necessary, the user may specify new FACS attribute codes to pass useful attributes not listed in the appendix. The User-Defined FACS Table records shall be used to document the meaning of these unique codes.

50.1.2.2.3.6 User-Defined FACS Table File. This optional file contains a list of any user-defined FACS codes used to encode feature attributes within the database.

50.1.2.2.3.6.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.2.3.6.2 User-Defined FACS Table Header Record. This mandatory record contains control information for a user-defined FACS table.

50.1.2.2.3.6.3 User-Defined FACS Table Entry Record. This mandatory record contains a single user-defined FACS code used to encode a feature attribute within the database. The number of these records corresponds to the count given in the header record.

50.1.2.2.3.7 Color Table File. This optional file contains a list of colors used by the model. Color table indices exist that point into this table.

50.1.2.2.3.7.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.2.3.7.2 Color Table Header Record. This mandatory record contains control information on the contents of the color table. It includes the number of colors and the color definition (RGB or HCV) used throughout the entire table.

50.1.2.2.3.7.3 Color Table Entry Record. This mandatory record contains a color value and description of a color used by the model. The color definition (RGB or HCV) used is defined in the Color Table Header Record.

50.1.2.2.3.8 Face-Based Texture Reference Table File. This optional file is used to define one method of placing a texture pattern on a polygon.

50.1.2.2.3.8.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.2.3.8.2 Face-Based Texture Reference Table Header Record. This mandatory record contains control information for the contents of the table.

50.1.2.2.3.8.3 Face-Based Texture Reference Record. The data contained in this record defines the transformation required to place a texture pattern on a polygon. The required texture map is expected to be present in a related SIF/HDI texture library file.

50.1.2.2.3.9 Vertex-to-Vertex Texture Reference Table File. This optional file is used to define one method of placing a texture pattern on a polygon.

50.1.2.2.3.9.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.2.3.9.2 Vertex-to-Vertex Texture Reference Table Header Record. This mandatory record contains control information on the contents of this table.

50.1.2.2.3.9.3 Vertex-to-Vertex Texture Reference Record. This mandatory record is used to define one method of placing a texture pattern on a polygon. This entails the mapping of texture pattern vertices to polygon vertices. The required texture map is expected to be present in a related SIF/HDI texture library file.

50.1.2.2.3.9.3.1 Texture Pattern Coordinates Subrecord. This subrecord lists the texture pattern coordinates that map to the polygon's vertices.

50.1.2.2.3.10 Model-Based Texture Reference Table File. This optional file is used to list a series of references to textures and their mapping parameters for model-based texturing. This type of texturing is used to project a single pattern onto multiple polygons (in different planes) simultaneously. This type of texturing can be conceptualized as the pattern being "shrink-wrapped" onto the model.

50.1.2.2.3.10.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.2.3.10.2 Model-Based Texture Reference Record. This mandatory record contains control information for the table.

50.1.2.2.3.10.3 Model-Based Texture Reference Record. This mandatory record is used to list a series of references to textures and their mapping parameters for model-based texturing. The required texture map is expected to be present in a related SIF/HDI texture library file.

50.1.2.2.3.11 Non-Mapped Texture Reference Table File. This optional file provides identification information for a referenced texture that is not mapped. This texture reference is used to reference textures that may be used but have not yet been mapped. It is intended for both specific and generic textures. It may be used to reference alternate textures as well.

50.1.2.2.3.11.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.2.3.11.2 Non-Mapped Texture Reference Table Header Record. This mandatory record contains control information for the table.

50.1.2.2.3.11.3 Non-Mapped Texture Reference Record. This optional record provides identification information for a referenced texture that is not mapped.

50.1.2.3 Culture Data. The purpose of this section is to define the detailed file, record, and field structure of the SIF/HDI culture data format.

50.1.2.3.1 Culture Data Encoding. The starting point for design of the SIF/HDI culture format was the SDBF Standard Simulator Data Base (SSDB), which is natural for a format intended to support interchange with the SSDB. The SSDB stores culture data in a vector graphics format (points, lines, and areals) in which vertices are expressed as 2-D or 3-D geographic coordinates. The format is conceptually comparable to DMA Digital Feature Analysis Data (DFAD), but considerably extended in terms of point precision and descriptive attributes supported.

a. The use of vector graphics to represent planimetric features is the general case in the simulator industry today, although some imagery-based simulators encode features as collections of pixels rather than as vectors. The pixel-based approach to feature classification is more properly handled within the photo texture segment of SIF/HDI.

b. A basic issue with vector culture formats is the degree of topology maintained within the data structure. At one extreme is the "spaghetti" Digital Landmass System (DLMS) DFAD format, which essentially treats every feature as an independent graphic entity. At the other extreme is a topological format, like DMA's Minitopo, which encodes detailed spatial relationships among features and graphic elements. In between are arc-node formats such as DMA Standard Linear Format (SLF), which support shared nodes and segments. The SSDB supports both "spaghetti" and arc-node data, but not a full topology. Arc-node format is a reasonable standard for SIF/HDI, as few (if any) simulator systems are configured to deal with topological data structures. Spaghetti data may be represented in an arc-node format but not vice versa. The digitizing conventions associated with the SIF/HDI approach to a limited feature topology are described within this standard.

c. DMA has announced plans to gradually migrate all of its standard vector products to a fully topological format called Vector Product Format (VPF), MIL-STD-600006 (DRAFT). Although VPF-like topologies are inefficient for real-time graphics rendering applications, they offer significant advantages for implementation of algorithms for automated feature thinning, filtering, aggregation, and the like. Therefore, it is expected that the simulator industry will gradually adopt topological data structures in their database generation systems. The SDBF intends to support topological data structures at a future time when the use of such data becomes more widespread among the user community. A preliminary design analysis indicates that the current SIF/HDI culture format would be able to support topological data structures by the addition of new optional record types corresponding to the Face Table and Ring Table of VPF. Since these optional records could be ignored when exchanging non-topological databases, they would not invalidate pre-existing SIF/HDI software and databases. However, it is recognized that this is a stopgap solution at best, and that the long-term approach must include the migration of SSDB culture into a truly VPF-compliant form.

d. Topology is also an issue across multiple representations of an area within various levels of detail (LODs). The SSDB supports multiple levels of detail (LODs) for any given area of coverage. These LODs represent general strata of feature resolution, roughly corresponding to feature resolutions of 100 meters, 30 meters, 10 meters, 3 meters, and 1 meter. As illustrated in Figure C-1, the SSDB format includes a system of inter-file pointers between alternate representations of features maintained at different LODs. These alternate representations reflect different levels of feature significance, aggregation, and generalization. An alternative approach to varying levels of feature detail used in many simulator databases is a system of embedded (merged) patches or "islands" of higher-resolution data within a lower-resolution background. The idea is to concentrate limited database processing capacities in areas of interest such as targets and navigation corridors. This type of structure is illustrated in Figure C-2.

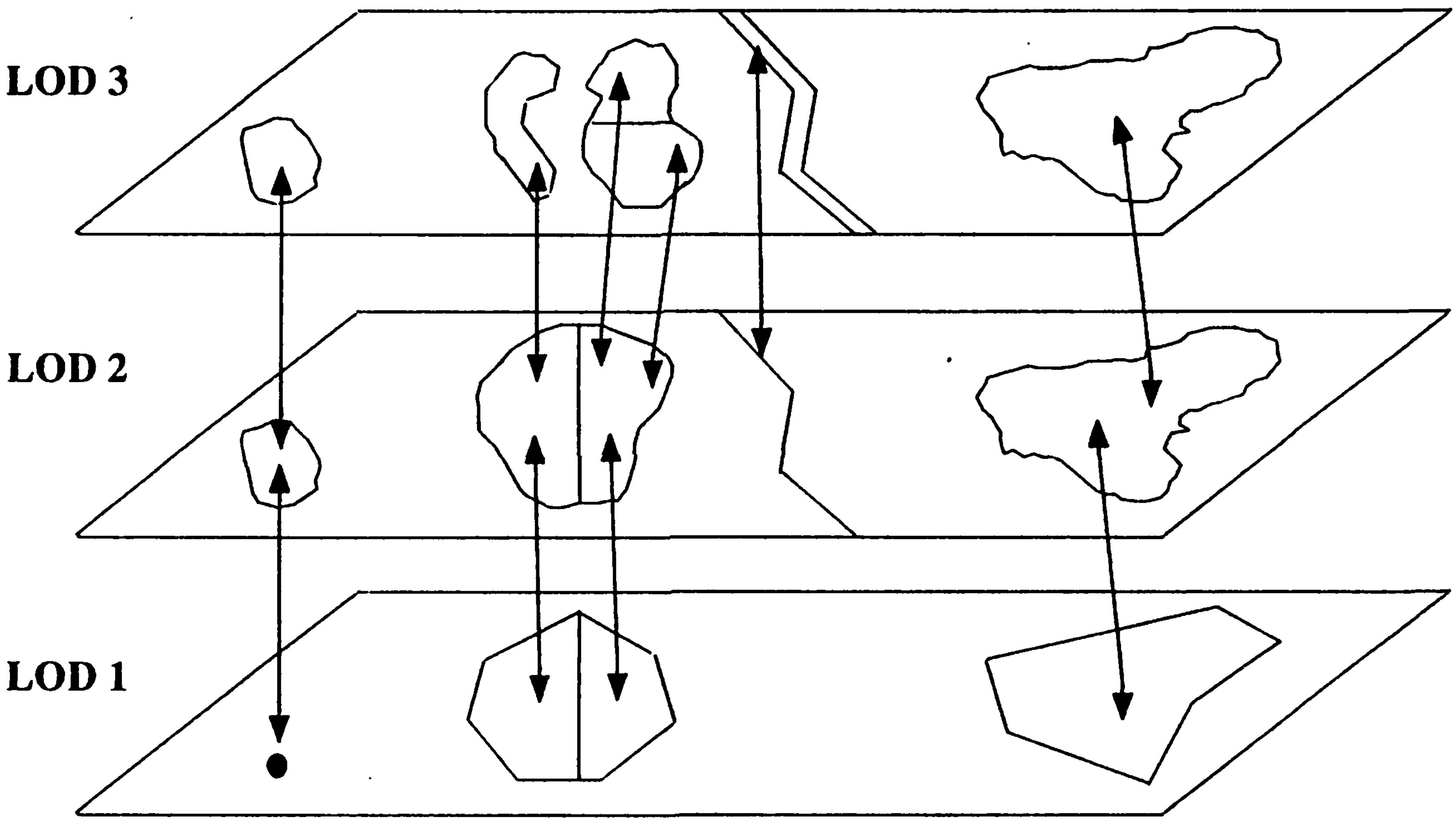


Figure C-1. Multi-Layer LOD Architecture.

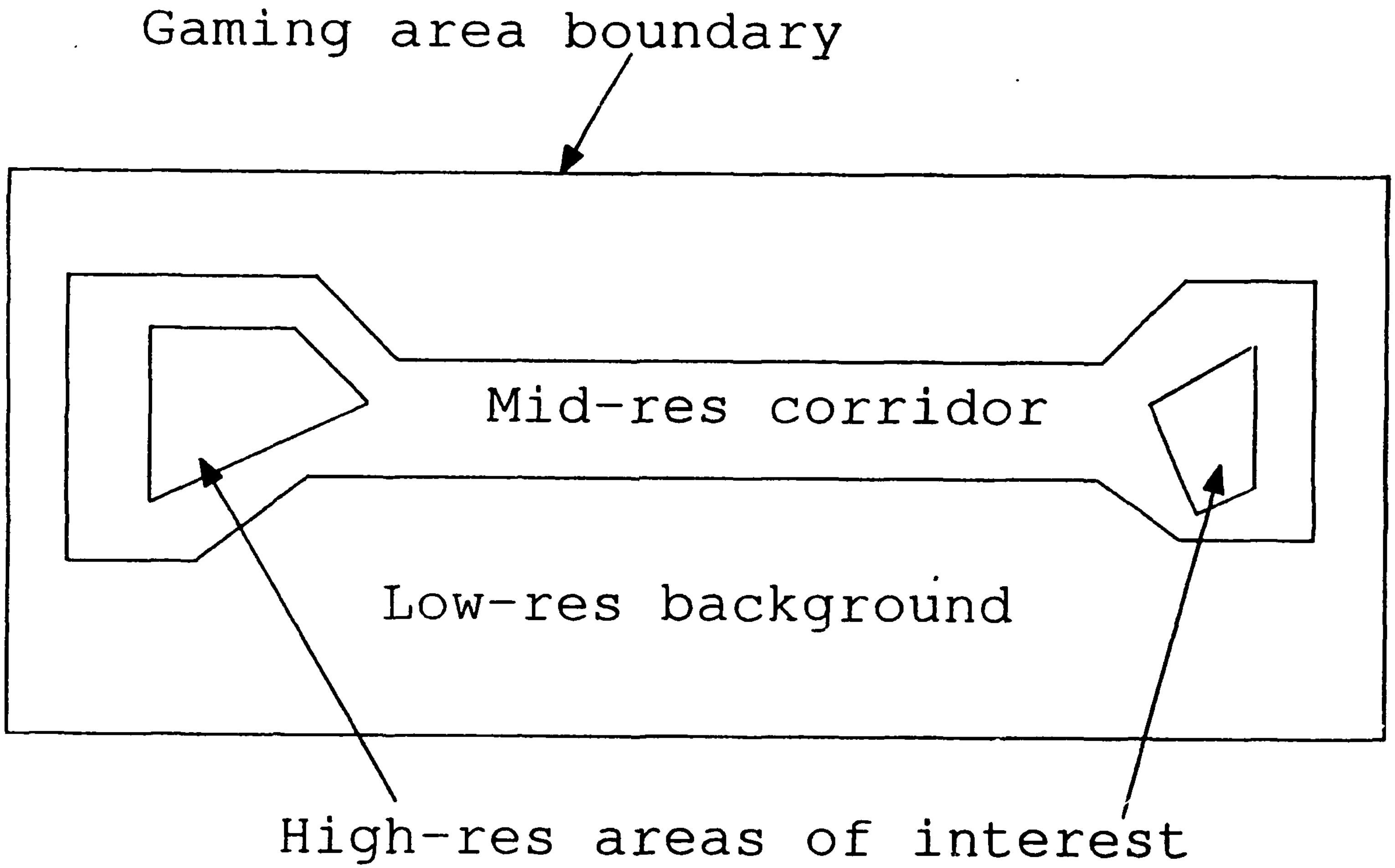


Figure C-2. Merged Embedded-Patch Architecture.

e. SIF/HDI supports geographic coordinate resolutions of a ten thousandth of an arc second. This resolution is needed to represent the shapes and relative positions of features included in high-resolution simulator database applications. As an option, SIF/HDI supports 3-D geographic coordinates as well as 2-D. The third dimension is elevation relative to Mean Sea Level (MSL) and is intended for use when representing terrain features such as point elevations, ridgelines, and contours. Theoretically, the 3-D culture format could be used to exchange polygonized terrain, as well as culture fragmented on underlying terrain polygons, but the SDBF SSDB does not directly store such data, and so this use is not supported within SIF.

f. In addition to representations of feature location and shape, the SIF/HDI format requires an approach for feature attribution. Attribute fields are needed to categorize each feature and encode characteristics useful for computer simulation of visual and sensor displays. In this regard, SIF/HDI deviates somewhat from the SSDB approach. The SSDB format includes a large number of fixed attribute fields, which take up space whether the attributes are known or not. To reduce the size of a potentially voluminous file, SIF/HDI uses a minimum number of fixed attributes which may expect to be captured in any simulator database. All other attributes are regarded as optional and are encoded in self-defining attribute records modeled after the DMA Feature Attribute Coding Standard (FACS). The FACS-like attribute coding scheme, which is also a part of the SSDB design, uses a generalized record structure which includes a code identifying what the attribute is, as well as giving the attribute value.

g. This section describes the fixed field portions of the different feature types supported by the SIF/HDI along with the optional fields that are to be populated via FACS records. These paragraphs first define the feature record, then identify the fixed fields, and lastly identify the optional fields. The translation of the textual description of these optional fields to FACS keywords will utilize either DMA FACS additional value codes (AV Codes) or SIF/HDI specific keywords. The keywords supported explicitly by SIF/HDI are documented in Appendix B to the GTDB Standard (MIL-STD-1820).

h. Similarly, SIF/HDI uses a Feature Descriptor Code modeled after the DMA FACS hierarchical feature codes to categorize features. (The SIF/HDI format also supports the Feature Identification Code from DMA DFAD.) The FACS-like approach gives each SIF/HDI user the flexibility to add new feature categories and attributes to SIF/HDI as the need arises. When exchanging databases between dissimilar systems, this is likely to prove highly useful in bridging the gap between different feature and attribute sets. SIF producers are encouraged to take advantage of this approach rather than be constrained by features and attributes explicitly supported by SIF at any given point in time; however, SDBF involvement needs to be a part of this process at all times, to ensure that FACS are utilized consistently.

i. The SIF/HDI implementation of features and attributes makes it possible to tag each feature and each of its attributes with a data source. This level of traceability will become important as SIF databases are exchanged and enhanced by different systems and users over time. The intent of such traceability is to allow the SDBF to make intelligent judgments on the relative quality and reliability of specific data items. As such, it is a critical component of any SIF data set, and cannot be overlooked by SIF producers.

j. An issue raised by the prospect of widespread database interchange is the degree of flexibility needed to support system-specific variations in such areas as coordinate systems, binary encoding formats, area blocking, and degree of transformation. The decision was made to establish standard conventions, to limit the amount of software development and testing needed to comply with the standard. Therefore, the header record descriptions in section 5.1 indicate conventions which must be followed when exchanging databases using SIF/HDI. Flexibility, when needed, is available through the use of the GTDB data base standard, rather than SIF.

50.1.2.3.1.1 Areal Feature Rules. (Self-explanatory.)

50.1.2.3.1.1.1 Background Feature. The purpose of the background feature is to provide default attribution for areas within the tile which are not explicitly defined by individual features.

50.1.2.3.1.1.2 Rendering Priority. All other areal features may be stored in an arbitrary sequence; i.e., the sequence does not imply the rendering priority.

50.1.2.3.1.1.3 Line Segments. The definition of segment ends may be arbitrary. For example, feature F1 is shown as consisting of four segments; feature F4 is a similar structure but has been defined as a single segment.

50.1.2.3.1.1.4 Shared Segments. For example, segment S5 (defined by vertices V5, V6, and V7) is a common segment shared by features F2 and F3.

50.1.2.3.1.1.5 Feature Traversal. (Self-explanatory.)

50.1.2.3.1.1.6 Closure. The last coordinate in the last segment must be identical with the first coordinate in the first segment.

50.1.2.3.1.1.7 Concave Features. The SDBF maintains non-convex areals within the SSDB. When requested by a user, the SDBF will decompose concave SSDB features into multiple convex features while creating a GTDB product, but does not offer such an option for SIF/HDI outputs.

50.1.2.3.1.1.8 Inside Segments. (Self-explanatory.)

50.1.2.3.1.1.9 Disjoint Polygons. (Self-explanatory.)

50.1.2.3.1.1.10 Non-redundant Vertices. (Self-explanatory.)

50.1.2.3.1.1.11 Vertex Ordering. The list positions serve as implicit keys used within the segment records to point to specific vertex records. The vertex coordinates within a vertex file may be arbitrarily sequenced. This means the recipient of a SIF/HDI database should not sort the vertex files, unless the vertex records have first been assigned explicit keys during input processing.

50.1.2.3.1.2 Linear Feature Rules. (Self-explanatory.)

50.1.2.3.1.2.1 Rendering Priority. Linear features may be stored in an arbitrary sequence; i.e., the sequence does not imply a rendering priority.

50.1.2.3.1.2.2 Line Segments. Within the SSDB, software is used to detect cases where one linear feature terminates at its intersection with another feature, and to force segment splitting at that point.

50.1.2.3.1.2.3 Segment Ends. (Self-explanatory.)

50.1.2.3.1.2.4 Shared Segments. (Self-explanatory.)

50.1.2.3.1.2.5 Directionality. Linear features which are visible or reflective only along one side of the feature (e.g., a dam) are referred to as uni-directional and may be flagged as such by use of the Directionality attribute.

50.1.2.3.1.2.6 Feature Traversal. (Self-explanatory.)

50.1.2.3.1.2.7 Disjoint Segments. (Self-explanatory.)

50.1.2.3.1.2.8 Non-contiguous Feature. (Self-explanatory.)

50.1.2.3.1.2.9 Non-redundant Vertices. (Self-explanatory.)

5.1.2.3.1.2.10 Vertex Ordering. The list positions serve as implicit keys used within the segment records to point to specific vertex records. The vertex coordinates within a vertex file may be arbitrarily sequenced. This means the recipient of a SIF/HDI database should not sort the vertex files, unless the vertex records have first been assigned explicit keys during input processing.

50.1.2.3.1.2.11 Feature/Segment Numbering. (Self-explanatory.)

50.1.2.3.1.3 Point Feature Rules. (Self-explanatory.)

50.1.2.3.1.3.1 Rendering Priority. Point features may be stored in an arbitrary sequence; i.e., the sequence does not imply a rendering priority.

50.1.2.3.1.3.2 Line Segments. (Self-explanatory.)

50.1.2.3.1.3.3 Vertex Sequence. (Self-explanatory.)

50.1.2.3.1.3.4 Disjoint Segments. (Self-explanatory.)

50.1.2.3.1.3.5 Coincident Segments. (Self-explanatory.)

50.1.2.3.1.3.6 Non-redundant Vertices. (Self-explanatory.)

50.1.2.3.1.3.7 Vertex Ordering. The list positions serve as implicit keys used within the segment records to point to specific vertex records. The vertex coordinates within a vertex file may be arbitrarily sequenced. This means the recipient of a SIF/HDI database should not sort the vertex files, unless the vertex records have first been assigned explicit keys during input processing.

50.1.2.3.1.3.8 Feature Numbering. (Self-explanatory.)

50.1.2.3.1.4 Point Light Feature Rules. (Self-explanatory.)

50.1.2.3.1.4.1 Rendering Priority. Point light features may be stored in an arbitrary sequence; i.e., the sequence does not imply a rendering priority.

50.1.2.3.1.4.2 Number of Vertices. As illustrated by feature F1, each point light feature shall be a segment consisting of one and only one vertex coordinate. (Features consisting of multiple point lights shall be encoded as point light string features.)

50.1.2.3.1.4.3 Coincident Segments. (Self-explanatory.)

50.1.2.3.1.4.4 Non-redundant Vertices. (Self-explanatory.)

50.1.2.3.1.4.5 Vertex Ordering. The list positions serve as implicit keys used within the segment records to point to specific vertex records. The vertex coordinates within a vertex file may be arbitrarily sequenced. This means the recipient of a SIF/HDI database should not sort the vertex files, unless the vertex records have first been assigned explicit keys during input processing.

50.1.2.3.1.4.6 Feature Numbering. (Self-explanatory.)

50.1.2.3.1.5 Point Light String Feature Rules. (Self-explanatory.)

50.1.2.3.1.5.1 Rendering Priority. Point light string features may be stored in an arbitrary sequence; i.e., the sequence does not imply a rendering priority.

50.1.2.3.1.5.2 Number of Vertices. (Self-explanatory.)

50.1.2.3.1.5.3 Non-linear Strings. (Self-explanatory.)

50.1.2.3.1.5.4 Parallel Strings. (Self-explanatory.)

50.1.2.3.1.5.5 Light Groups. (Self-explanatory.)

50.1.2.3.1.5.6 Vertex Sequence. (Self-explanatory.)

50.1.2.3.1.5.7 Coincident Segments. (Self-explanatory.)

50.1.2.3.1.5.9 Vertex Ordering. The list positions serve as implicit keys used within the segment records to point to specific vertex records. The vertex coordinates within a vertex file may be arbitrarily sequenced. This means the recipient of a SIF/HDI database should not sort the vertex files, unless the vertex records have first been assigned explicit keys during input processing.

50.1.2.3.1.5.10 Feature Numbering. (Self-explanatory.)

5.1.2.3.1.6 Model Reference Rules. (Self-explanatory.)

50.1.2.3.1.6.1 Number of Vertices. Note that there is no requirement that the model reference coordinate exactly correspond to any vertex in the culture vertex files.

50.1.2.3.1.6.2 Model Reference Table. (Self-explanatory.)

50.1.2.3.1.6.3 Multiple References. In general, it is expected that there will be a one-to-one substitution of a model for a feature. However, it is possible for a single complex model to be substituted for multiple culture features. Models may be substituted for any combination of areal, lineal, point, point light, and point light string features.

50.1.2.3.1.6.4 Multiple Models. This indicates that there is a choice of models available for that feature. Note that each model reference is permitted to have a slightly different placement coordinate due to differences in model geometries.

50.1.2.3.1.6.5 Placement Coordinate. The default for SDBF-generated model references is 2-D.

50.1.2.3.1.6.6 Table ID. (Self-explanatory.)

50.1.2.3.1.7 Superfeature Rules. The superfeature was created to combine or aggregate like features into larger homogeneous data groups. The smaller features, or child features, which make up the superfeature are tied together through pointer references.

50.1.2.3.1.7.1 Child Feature References. A child feature is one of several features which defines a superfeature. There are no restriction on the types or dimensions of the child feature referenced to include grouping of unlike features (areals with linear, linears with points, 2-D with D-3, etc.). Additionally, child features can belong to more than one superfeature.

5.1.2.3.1.7.2 "Aggregate" Feature References. A 2-D polygon is considered an aggregate feature of the 3-D superfeature when the 2-D polygons' spatial boundaries corresponds to the boundaries of the 3-D superfeature. A possible application is to then replace the 3-D superfeature with the 2-D aggregate version or vise versa.

5.1.2.3.1.7.3 Additional References. A superfeature can be a child feature to a larger superfeature. This larger superfeature can consist of both individual features and superfeatures.

50.1.2.3.2 Culture Section Structure. Each SSDB culture manuscript is described by a group of files made up of variable-length keyword records. These files consist of culture features that include both geometry and attribute information. A valuable feature of the existing SSDB format is its support for a FACS-like attribute coding scheme. Modeled after DMA's Feature Attribute Coding System, it is a system of self-defining feature attributes, which gives total flexibility to add new attributes to SIF/HDI as the need arises. The user-defined FACS mechanism should be used only when absolutely necessary. In order to be compliant with this specification, external systems creating a SIF/HDI database shall use the attributes already defined in this specification wherever possible. There is much extended use of FACS attributes to avoid the overhead of many fixed attributes which are rarely populated. FACS attribution has already allowed for the definition of a large number of new FACS attributes not currently in the GTDB. Simple table structures have been employed to support flexible FACS attribution, colors in either red-green-blue (RGB) or hue-chroma-value (HCV) formats, and FID/FDC cross-referencing. To allow for the greatest amount of readability along with space saving techniques, SIF/HDI uses compressed ASCII files and binary data files.

50.1.2.3.2.1 Database Header File. This mandatory file contains control information describing the database contents.

50.1.2.3.2.1.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.3.2.1.2 SIF/HDI Culture Database Header Record. This mandatory record contains control information pertinent to the entire culture database.

50.1.2.3.2.1.3 Data Source Table Record. This mandatory record is used to document the data source(s) used to populate and/or update the culture files being transmitted.

50.1.2.3.2.1.4 Accuracy Region Record. This optional record contains an array defining data accuracy standards for multiple regions within the area of coverage of a data source. Regions of differing accuracy are possible when a data source is itself a composite product generated from varying original sources.

50.1.2.3.2.2 Tile Information File. This mandatory file contains coverage information for the culture database. Information within this file consists of overall coverage per tile of the database, and information abount high resolution islands.

50.1.2.3.2.2.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.3.2.2.2 Tile Header Record. (Self-explanatory.)

50.1.2.3.2.2.3 Data Resolution Identifier Record. Within each tile, there is support for identification of embedded patches of higher-resolution data called "islands." Many simulator databases insert and feather such patches for high-interest areas such as waypoints and targets. Since the SSDB maintains different LODs separately, these patches must be extracted from the surrounding data when stored in the SSDB. To support this capability, the Data Resolution Identifier includes fields for identifying the boundaries of any island.

50.1.2.3.2.3 Two-D Coordinate File. This file is optional only if a Three-D Coordinate File is included in the SIF/HDI culture database. This file contains latitude and longitude coordinates for culture segments. A Two-D Coordinate File is based at the tile level.

50.1.2.3.2.3.1 2-D Coordinate Record. These records define the vertices of culture segments. Currently the SSDB supports data resolutions of thousandths of arc seconds, so the SIF user should be aware that culture data stored in the SSDB will not maintain the accuracy of one ten thousandths of an arc second.

50.1.2.3.2.4 Three-D Coordinate File. This file is optional only if a Two-D Coordinate File is included in the SIF/HDI culture database. This file contains latitude, longitude and elevation coordinates for culture segments. A Three-D Coordinate file is based at the tile level.

50.1.2.3.2.4.1 3-D Coordinate Record. Currently the SSDB supports data resolutions of thousandths of arc seconds, so the SIF user should be aware that culture data stored in the active SSDB will not maintain the accuracy of one ten thousandths of an arc second. One or more of the coordinate records may be used to define the vertices of a culture segment.

50.1.2.3.2.5 FACS Table File. This optional file serves two purposes: (1) to minimize space by eliminating redundant feature attribute assignments, and (2) to allow expandability of supported attributes. A FACS Table Index pointing to the appropriate table entry can be found in several records. If there are no features that use additional descriptors, this table may be omitted from the SIF transmittal for the culture tile.

50.1.2.3.2.5.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.3.2.5.2 FACS Table Header Record. This mandatory record supplies top level information about the records contained within the FACS table file.

50.1.2.3.2.5.3 FACS Table Entry Record. The FACS Table entry in the FACS Table is composed of two control fields and a variable number of FACS Attribute records.

50.1.2.3.2.5.3.1 FACS Attribute Subrecord. This record contains a FACS (Feature Attribute Coding Standard) attribute value (AV Code plus value) associated with one or more features. Multiple records may be used to store FACS attributes, if required. This record should be used to pass any attributes for a feature in addition to the "core" attributes stored in the parent feature record. Such attributes may include various physical, cultural, or sensor-response characteristics as may be needed to support a simulation. The keywords supported explicitly by SIF/HDI are documented in a Appendix B. When necessary, the user may specify new FACS attribute codes to pass useful attributes not listed in the appendix. The User-Defined FACS Table records shall be used to document the meaning of these unique codes.

50.1.2.3.2.6 User-Defined FACS Table File. This optional file contains a list of user-defined FACS codes used to encode feature attributes within the database.

50.1.2.3.2.6.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.3.2.6.2 User-Defined FACS Table Header Record. This mandatory record supplies top level information about the records contained within the FACS table file.

50.1.2.3.2.6.3 User-Defined FACS Table Entry Record. This mandatory record contains one entry in the User-Defined FACS Table.

50.1.2.3.2.7 Color Table File. An optional color table will accompany the feature file. The color table may be used to define the colors used within the feature records.

50.1.2.3.2.7.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.3.2.7.2 Color Table Header Record. This mandatory record supplies top level information about the records contained within the color table file.

50.1.2.3.2.7.3 Color Table Entry Record. The color in each entry of the color table may be defined as either Red/Green/Blue or as Hue/Chroma/Value. The Color Definition Type field in the Color Table Header Record will indicate whether RGB or HCV is being used.

50.1.2.3.2.8 FID/FDC Cross-Reference Table File. This optional file provides a method for cross-referencing a user's Feature Identification Code (FID) with the SDBF Feature Descriptor Code (FDC).

50.1.2.3.2.8.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.3.2.8.2 FID/FDC Cross-Reference Header Record. This mandatory record supplies top level information about the records contained within the FID/FDC Cross Reference Table file.

50.1.2.3.2.8.3 FID/FDC Cross-Reference Table Entry Record. The FID/FDC Cross Reference Table provides a method for cross-referencing a user's Feature Identification Code (FID) with the SDBF Feature Descriptor Code (FDC).

50.1.2.3.2.9 Global-Based Texture Reference Table File. The Global-Based Texture Reference Table provides a method for cross-referencing a culture feature with a texture map that has been mapped to it.

50.1.2.3.2.9.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.3.2.9.2 Global-Based Texture Reference Table Header Record. This mandatory record supplies top level information about the records contained within the Global-Based Texture Reference Table file.

50.1.2.3.2.9.3 Global-Based Texture Reference Record. The Global-Based Texture Reference Record provides specific parameters for texture mapping of culture features.

50.1.2.3.2.10 Non-Mapped Texture Reference Table File. The Non-Mapped Texture Reference Table provides a method for cross-referencing a culture feature with a texture map that has not yet been mapped to it. It is intended for generic textures only. (They can be mapped to individual homogeneous culture features. Geospecific textures are associated with heterogeneous geographic areas).

50.1.2.3.2.10.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.3.2.10.2 Non-Mapped Texture Reference Table Header Record. This mandatory record supplies top level information about the records contained within the Non-Mapped Texture Reference Table file.

50.1.2.3.2.10.3 Non-Mapped Texture Reference Record. The Non-Mapped Texture Reference Record provides identification of referenced textures that have not been mapped to the referencing culture feature.

50.1.2.3.2.11 Model Reference Table File. The Model Reference Table provides a method for cross-referencing one or more culture feature(s) with a model, or including model references for a culture tile that are not tied to specific features.

50.1.2.3.2.11.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.3.2.11.2 Model Reference Header Record. This mandatory record supplies top level information about the records contained within the Model Reference Table file.

50.1.2.3.2.11.3 Model Reference Table Entry Record. The Model Reference Table Entry Record has two uses in the SSDB. First, models may be introduced into the SSDB culture not as direct substitutions for point, line, or areal features, but rather to represent ground clutter (e.g., trees, sagebrush) or to provide realistic detail (e.g., runway markings). The Model Reference Table Entry Record is used to position and orient such models within a culture tile. Second, even when a model is intended as an optional direct substitution for a feature, there will be cases when special scaling and orienting instructions may be needed to make the model look right. The Model Reference Record may be used to provide offset vectors defining specific parameters for model placement and orientation for the particular use.

50.1.2.3.2.12 Superfeature File. The superfeature file provides a method for aggregating culture features into homogeneous groupings.

50.1.2.3.2.12.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.3.2.12.2 Superfeature Header Record. This mandatory record supplies all information for a given superfeature. This information includes a list of all features that are grouped into the superfeature, a mechanism to identify whether the referenced feature(s) are children or "aggregate" feature(s). There is a description field contained with each superfeature to allow for a user defined grouping methodology.

50.1.2.3.2.13 Feature File. This mandatory file contains various records describing all the features contained in the culture tile.

50.1.2.3.2.13.1 Feature Record. (Self-explanatory.)

50.1.2.3.2.13.2 SIF File Identifier Record. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.3.2.13.3 Manuscript Header Record. This mandatory record contains control information applying to the feature data file.

50.1.2.3.2.13.4 Areal Feature Record. There will be an Areal Feature Record describing each areal feature within a culture file. An areal feature is an object defined spatially as a closed contour (polygon). By convention, every culture manuscript in the SIF must have at least one areal feature defining the "background" or default feature attributes.

50.1.2.3.2.13.5 Linear Feature Record. There will be a Linear Feature Record describing each linear feature within a culture file. A linear feature is an object defined spatially by one or more connected line segments which do not close to form a polygon.

50.1.2.3.2.13.6 Point Feature Record. There will be a Point Feature Record describing each point feature within a culture file. A point feature is an object defined spatially by either a single coordinate or by a sequence of discrete but logically connected points.

50.1.2.3.2.13.7 Point Light Feature Record. There will be a Point Light Feature Record describing each point light feature within a culture file. Point light features are light emitting objects represented spatially by a single coordinate. They contain several attributes necessary for describing a light emitter such as the light lobe parameters, cycle rate, light type, and intensity.

50.1.2.3.2.13.8 Point Light String Feature Record. There will be a Point Light String Feature Record describing each point light string feature within a culture file. Point light strings are a sequence of discrete but logically connected light emitters. They are similar in data structure to a point light, but they have several additional attributes required for describing the shape and placement of the string and the lights within it.

50.1.2.3.2.13.9 Culture Segment Pointer Record. This record contains an array of segment list pointers defining the geometry of a particular feature.

50.1.2.3.2.13.10 Model Reference Pointer Record. The Model Reference Pointer Record is used to identify a Model Reference Record contained within the Model Reference Table.

50.1.2.3.2.13.11 Microdescriptor Record. This optional record contains a microdescriptor associated with a feature.

50.1.2.3.2.13.12 Feature Continuation Record. This optional record contains a pointer from a feature in the current culture manuscript file to a continuation feature in an adjoining manuscript.

50.1.2.3.2.13.13 FACS List Pointer Record. This optional record contains a pointer into the FACS table that is maintained for the current database tile. One or more of these records may be used to store pointers to additional entries into the FACS Table for the current feature. This record should be used to identify any additional attributes for a feature in addition to the "core" attributes stored in the parent feature record. Such attributes may include various physical, cultural, or sensor-response characteristics as may be needed to support a simulation.

50.1.2.3.2.13.14 <u>Texture Reference Pointer Record</u>. This optional record contains a texture map reference pointer associated with a feature.

50.1.2.3.2.13.15 <u>LOD Cross Reference Record</u>. This optional record is used to provide the information required to link together different representations of a feature. When multiple levels of detail (LOD) data are transmitted via the SIF/HDI (layered culture data), rather than one merged level of detail, any given feature could exist at a coarse LOD and a finer LOD. For example, a feature at LOD 1 can point to one or more features at LOD 2, but a feature at LOD 2 can only point at one and only one feature at LOD 1. The record format is the same for higher or lower LOD cross references except for the keyword identifier which is used to identify whether the cross reference is a higher LOD or lower LOD reference.

50.1.2.3.2.14 <u>Segment File</u>. This mandatory file contains various records describing all the segments contained in the culture tile.

50.1.2.3.2.14.1 <u>SIF File Identifier Record</u>. This mandatory record contains identifiers indicating the data section and type of file.

50.1.2.3.2.14.2 <u>Segment Header Record</u>. A segment is a logically connected string of vertex coordinates used to define a feature, or a part of a feature. A segment consisting of a single coordinate may be used to define the location of a point or point light feature. A linear or areal feature may be split into an arbitrary number of segments, each containing an arbitrary number of coordinates. Typically, a feature is broken into segments at topologically significant node points such as the intersection between features, but segmentation may also be a purely arbitrary artifact of the digitization process. The actual vertex coordinates making up a segment are stored in separate Coordinates Records. The total number of coordinates making up a segment is given in the Number of Coordinate Pairs field. Each Segment Header Record serves as a bi-directional pointer relating one or more feature records with each segment. In the primary direction, the Segment Header Records are referenced by the Culture Segment Pointer Records associated with the various feature records. A segment may be shared by two or more features, in which case each of those features will point to the shared segment header. In the opposite direction, each Segment Header Record will contain one or more backpointers, which are pointers from the segment back to the feature(s) which reference that segment. This bi-directionality makes it possible for software, given a feature, to extract all segments making up that feature, and, given a segment, to identify all features making use of that segment.

50.1.2.3.2.14.3 <u>Vertex List Pointer Record</u>. This mandatory record contains an array of pointers into either the 2-D Segment Coordinates File or the 3-D Segment Coordinates file based on the value in the 2-D/3-D Coordinates Flag field in the parent Segment Header Record.

50.1.2.3.2.14.4 <u>Segment Backpointer Record</u>. This mandatory record contains an array of segment backpointers.

50.1.2.4 <u>Gridded Data</u>. This section defines the detailed file, record, and field structure of the SIF/HDI gridded data format. This format is used to represent both the terrain and the texture components of a SIF/HDI database.

50.1.2.4.1 Gridded Data Encoding. SIF/HDI treats several forms of gridded simulator data in a common manner. These data types include imagery-based and generic forms of photo texture as well as gridded terrain data. Although photo texture and terrain have traditionally been maintained in different formats, they have fundamental architectural simularities in that both are grids. What varies are the characteristics described within the grid as well as the specific geometry of each grid.

a. SIF/HDI builds upon an existing imagery standard, the National Imagery Transmission Format (NITF), which was developed by the U.S. intelligence community to be independent of specific image handling workstations and systems. NITF is extremely flexible by design and accomodates traceability, security information, compressed imagery, encrypted imagery, multi-band imagery and image filter conditions, as well as supporting look-up tables. This standard is intended to support exchange of image data among a wide range of vendors and users having potentially different and incompatible internal formats.

b. Within the simulator database community, there are several alternative methods for representing terrain which have their pros and cons. The most widely used digital terrain product, Defense Mapping Agency (DMA) Digital Terrain Elevation Data (DTED), is a gridded format, in which elevation values are assigned to a matrix of positions (terrain "posts") defined by fixed arcs of latitude and longitude. Almost all the actual DTED produced by DMA is in 3-second arc spacing, which is roughly equivalent to 100 meters of ground distance at the mid-latitudes. This data is often colloquially referred to as "100-meter DTED." The Standard Simulator Data Base (SSDB) also stores terrain data in a gridded format, but supports a wider range of post spacings.

c. As an alternative to a gridded structure, many simulator systems polygonize the terrain. A polygonized format, like a triangulated irregular network, can be much more efficient than a grid in representing terrain. It also simplifies real-time image rendering calculations. Thus, although the SDBF maintains terrain as a grid in the SSDB, it offers the option of having gridded terrain, polygonized terrain, or both, within its primary output product, the Generic Transformed Data Base (GTDB).

d. A third alternative is to represent terrain using vector graphic data structures (points, lines, and polygons) normally associated with culture data. The advantage here is the potential for greater precision in representation of important terrain features such as ridgelines and point elevations.

e. In order to support general sharing of terrain data, SIF/HDI is primarily a gridded format but will also support vector terrain features. Gridded terrain is more of a common denominator than triangulated networks or vector terrain. Even when a simulator uses polygonized terrain in its real-time database, a gridded representation is typically used as a higher-resolution starting point within the database generation system. The most commonly cited drawbacks to the gridded format are its voluminousness at high resolutions, and its inability to precisely capture terrain vertices at lower grid resolutions. To address these concerns, the SIF/HDI format supports variable post spacings and vector terrain features.

f. Variable post spacings will permit sharing of data at high resolutions without forcing everyone always to capture data at the highest resolution. Each terrain file must have a consistent grid spacing, as defined in the header record for that file, but the choice of grid spacing is unconstrained and left to the producer. The size (geographic extent) of each terrain file may also be user defined, thereby minimizing "empty" terrain posts caused by fixed file sizes.

g. Vector terrain features will support representation of very precise terrain relief. Since a general vector data structure is used to represent feature data, it was decided to include terrain features within the feature data component of SIF/HDI and not the terrain component per se. Therefore, this section documents only the SIF/HDI approach to gridded terrain. The feature data format is defined in a separate section of this standard.

50.1.2.4.2 <u>Gridded Data Section Structure</u>. (Self-explanatory.)

50.1.2.4.2.1 <u>Basic NITF structure</u>. The original intended application for NITF was electronic transmission of exploited intelligence imagery. Thus, the headers are structured as message headers, and the data are structured as a series of non-destructive overlays. (A non-destructive overlay is information which can be displayed over an underlying image but which has not been physically merged as data pixels within the image.) Each message is expected to contain one exploited image along with overlays and amplifying information. An NITF Header is required at the beginning of each message. NITF supports six classes of data following each header: image, symbol, label, text, audio, and non-static presentation information (NPI). All data classes are optional, with each instance defined by an NITF Sub-Header. There is provision for user-defined headers and sub-headers to support other data types. SIF/HDI takes advantage of this feature for data such as terrain. NITF has also reserved space for extended headers and sub-headers to support future expansion of the standard.

50.1.2.4.2.2 <u>SIF/HDI application-specific features</u>. SIF/HDI uses the user-defineable aspects of NITF to define a new data class to represent terrain. The overall data architecture of an NITF "message" is to begin with an NITF Header, and then follow it with all instances of each type of data in turn.

50.1.2.4.3 <u>Gridded Data File Structures</u>. (Self-explanatory.)

50.1.2.4.3.1 <u>NITF Header File</u>. This mandatory file represents the NITF Header data. In order to ensure forward compatibility with NITF as it evolves independently from SIF/HDI, all SIF-specific enhancements have been placed within the User Defined Header Data field (which can contain up to 99,999 characters). The reader should consult the SIF/HDI and SIF/DP Data Dictionary (Appendix A of this standard) and the NITF documentation for an explanation of all fields which includes field size and range of values. The format of this file is the same as that in the NITF standard. It is provided here for completeness and convenience.

50.1.2.4.3.1.1 SIF/HDI User Defined Header Data. Within the NITF Header, the SIF/HDI extensions provide counts of SIF/HDI-specific data class files to follow. These are the terrain data files. It also includes information for the texture images: a list of all the tie points and generic texture association data.

50.1.2.4.3.2 Terrain Sub-Header File. This file is mandatory for each terrain tile. The Terrain Sub-Header file is used to describe the terrain elevation data file which immediately follows the header. It is a totally SIF/HDI-specific extension to NITF. However, in order to ensure forward compatibility with NITF as it evolves independently from SIF/HDI, the Terrain Sub-Header Record is identical to the Image Sub-Header Record with a few exceptions. Consistent with NITF, all fields in the Terrain Sub-Header are encoded as ASCII characters, including numeric values.

a. NITF limits the number of pixels (terrain posts) per image to 4096 in the horizontal direction and 7700 in the vertical. SIF/HDI does not observe this limitation.

b. In the case of terrain, SIF/HDI does not observe the limit of 16 bits per pixel. For SIF/HDI terrain, either 16 or 24 bits shall be used consistently throughout a terrain manuscript to support 1.0 meters of elevation precision or 0.01 meters of elevation precision, respectively.

c. Image Geographic Location is limited in NITF to the nearest second in latitude and longitude. This may not be good enough for very high-resolution terrain patches, since 1 arc second spacing represents roughly 30 meters of ground resolution. Therefore, the format of the Terrain Geographic Location field has been changed, from NITF's four corner coordinates expressed to the nearest arc second, to four coordinates expressed in units of thousandths of arc seconds. Also, while the NITF Image Coordinate System can be Universal Transverse Mercator (UTM), geodetic/geographic, geocentric, or none, the SIF/HDI Terrain Coordinate System must be geodetic/geographic.

d. Except for SIF/HDI specific fields which are found in the User Defined Terrain Data area, the reader should consult the SIF/HDI and SIF/DP Data Dictionary and the NITF documentation of equivalent fields within the Image Sub-Header for an explanation of field size and range of values.

50.1.2.4.3.2.1 SIF/HDI User Defined Terrain Data. Within the SIF/HDI Terrain Sub-Header, the User-Defined extensions are needed primarily to better document the sources of data used to arrive at the terrain elevations.

50.1.2.4.3.3 Terrain Data File. This file is mandatory for each terrain tile. The SIF/HDI terrain data format is a sequence of elevation values, where the size and geometry of the grid positions has been defined in the preceding Terrain Sub-Header. The data sequence is always left to right and top to bottom. Note that this sequence is different from the sequence used within the GTDB, in that SIF/HDI is based upon NITF, whereas GTDB is based on DTED.

50.1.2.4.3.4 Image Sub-Header File. This file is mandatory for each image. The Image Sub-Header is used to describe the texture image which immediately follows the header. In order to ensure forward compatibility with NITF as it evolves independently from SIF/HDI, all SIF-specific enhancements have been placed within the User Defined Image Data field. Consistent with NITF, all fields in the Image Sub-Header are encoded as ASCII characters, including numeric values.

a. The SIF/HDI implementation of the NITF standard has some exceptions to the standard. These differences affect the Image Title Field, Image Coordinate System Field, the Image Geographic Location Field, the use of Look Up Tables, and the image size.

b. The length and format of the Image Title field is left intact in the SIF/HDI implementation; however, it should be noted that SIF/HDI makes use of only the first 20 characters of the 80 character field, with the remaining 60 characters ignored. When implementing SIF/HDI, one should use only the first 20 characters. If a more lengthy title or description is needed, it can be transferred via the 80-character Texture Description Field in the SIF/HDI User Defined Image Data section.

c. Image Geographic Location is limited in NITF to the nearest second in latitude and longitude. This may not be good enough for very high-resolution imagery, since 1 arc second spacing represents roughly 30 meters of ground resolution. Therefore, the format of the Image Geographic Location field has been changed, from NITF's four corner coordinates expressed to the nearest arc second, to four coordinates expressed in units of thousandths of arc seconds. It is noted that they may not be an exact outline of the image. A more accurate outline is found in the Texture Footprint Data in the User-Defined Image Data.

d. NITF limits the number of pixels per image to 4096 in the horizontal direction and 7700 in the vertical, with a maximum of 16 bits per pixel per band. SIF/HDI does not observe this limitation.

e. The reader should consult the SIF/HDI and SIF/DP Data Dictionary and the NITF documentation for an explanation of all fields (to include field size and range of values) except for the SIF/HDI specific fields found in the SIF/HDI User Defined Image Data area.

50.1.2.4.3.4.1 SIF/HDI User Defined Image Data. Within the NITF Image Sub-Header, the SIF/HDI extensions are needed primarily to better document the sources of data and the processing used to arrive at the texture. The SIF/HDI-specific data is specified later in this section by the field name, the size in bytes, valid values for the field, and a usage type for the field. The usage type column provides an 8-character value for each field. Each character represents one of the usage type codes previously specified: Required (R), Optional (O), Conditional (C), and Null (N). (Note that the Required, Optional, and Null fields must be provided; the terms "Required", "Optional", and "Null" refer to the existence of valid data within those data fields.) Each of the eight usage type codes represents a certain type of texture. Each of the positions in the 8-character value corresponds to the following eight texture library types, respectively:

Stage 1 Specific Areal
Stage 2 Specific Areal
Stage 3 Specific Areal
Stage 1 Specific Model
Stage 2 Specific Model
Stage 3 Specific Model
Generic
SMC/FDC

50.1.2.4.3.4.1.1 Stage 1 Specific Areal. (Self-explanatory.)

50.1.2.4.3.4.1.2 Stage 2 Specific Areal. (Self-explanatory.)

50.1.2.4.3.4.1.3 Stage 3 Specific Areal. (Self-explanatory.)

50.1.2.4.3.4.1.4 Stage 1 Specific Model. (Self-explanatory.)

50.1.2.4.3.4.1.5 Stage 2 Specific Model. (Self-explanatory.)

50.1.2.4.3.4.1.6 Stage 3 Specific Model. (Self-explanatory.)

50.1.2.4.3.4.1.7 Generic. (Self-explanatory.)

50.1.2.4.3.4.1.8 SMC/FDC. (Self-explanatory.)

50.1.2.4.3.4.2 Field Structure. (Self-explanatory.)

50.1.2.4.3.5 Image Data File. This file is mandatory for each image. The SIF/HDI image data format is a sequence of pixel values, where the size and geometry of the pixels have been defined in the preceding Image Sub-Header. The data sequence is always left to right and top to bottom. Per NITF, the format supports either Band Sequential or Band Interleaved approaches when multi-band imagery is involved. As stated in the NITF standard document:

a. "Image data within a block shall be formatted on a row by row basis, from left to right along each row or line, and from the top of the block to the bottom, down the rows. Data shall begin with the N bits of pixel (0,0) (the first row, first column) of the first block." The NITF image coordinate system starts with (0,0) at the upper left corner of the image, with the first coordinate increasing from top to bottom, and the second coordinate increasing from left to right. "The N bits of each pixel shall be in order beginning with the most significant bit (MSB) and ending with the least significant bit (LSB). This is followed by the N bits of data for pixel (0,1), which is the first row, second column of the first block. The N bits of data for pixel (1,0) (the first column of the second row) of the first block shall follow the last pixèl of the first row of the first block. The MSB of data for the first pixel of the first line of the second block shall follow the LSB of data for the last pixel of the last line of the preceding block. The end of the image data shall be LSB of the N bits of the last pixel, last row, last block of the last band."

b. "In Sequential Image Mode (i.e., Band Sequential), all of the blocks of the first band are followed by all of the blocks by the second band, and so on. Thus, the first block of the first band is followed by the data for the second block of the first band. The last block of the first band is then followed by the first block of the second band. In Interleaved Image Mode (i.e., Band Interleaved or Pixel Interleaved), the first block of the first band is followed by the first block of the second band which is then followed by the first block of each subsequent band. The first block of the last band is followed by the second block of the first band, and so on."

c. While the storage of visual textures in the Image Data File is straight-forward, the storage method for non-visual textures, namely SMC/FDC codes, is less apparent; thus, an explanation follows. It is expected that an SMC/FDC texture shall contain many texels with the same values (i.e., homogeneous areas) and that a Look-up Table (LUT) is the most cost-effective way of storing this data. If an LUT is used, the entries shall be entirely ASCII with a total length of seven bytes: the first two representing the SMC, while the following five represent the FDC value.

50.1.3.2.5 Data Quality. This section defines the quality requirements for the information contained in a SIF data set.

50.1.3.2.5.1 General. The subparagraphs hereto apply to all sections of the SIF data set.

50.1.3.2.5.1.1 Boundary Integrity. It must be ensured that information does not fall outside the specified boundaries. It also must be ensured that the information contained within the boundaries is consistent and non-redundant.

50.1.3.2.5.1.2 Data Values. All data values must fall within the defined ranges, whether defined by this standard or user-defined.

50.1.3.2.5.1.3 Source Traceability. Since most SIF data will have been derived from other types of source material, it is necessary to preserve this heritage in order to know how good the SIF data set itself is. Although many external DBGSs do not record this information explicitly, it is, in fact, known to the operators of that equipment, and is therefore available for inclusion in the SIF data set, even if it requires manual input.

50.1.3.2.5.1.4 Levels of Detail. Since the SSDB is stored internally in multiple levels of detail, it is most efficient to receive SIF information in that form. It is unlikely that the SDBF will have adequate resources to perform the segregation of single-layer, multiple-resolution SIF data sets into the multiple-LOD form needed internally; therefore, any external data bases which exist as single layers should be broken out as they are converted into SIF, not after.

50.1.3.2.5.1.5 Post-Acceptance SIF Generation. Under any training simulator contract, the quality of the primary deliverable data base (the real-time data base) continually varies throughout the development of the system. Even during acceptance testing, anomalies are noted and corrected. As a result, the data base cannot actually be considered "correct" until after it is accepted by the Government. To avoid having to make changes twice, the corresponding SIF data set should not be generated until after the acceptance of the real-time data base, when its content has finally stabilized.

50.1.3.2.5.2 Culture Data Section. (Self-Explanatory.)

50.1.3.2.5.2.1 Capture Criteria. Capture criteria are used to define how information is allocated to the various levels of detail in the SSDB. In the case of culture, capture criteria are nominally correlated to hardcopy maps, and the presence or absence of a particular feature is based upon its occurrence in the corresponding map. To minimize the impact of integrating culture from externally-produced SIF data sets into the SSDB, the criteria by which this allocation is made need to be the same as those used internally by the SDBF.

50.1.3.2.5.2.2 Derivative Areal Features. Decomposed (or fragmented) features can add a significant amount of additional processing to consumers of a data set, most of whom will need to reassemble the individual polygons into something resembling the original feature. This is necessary because feature decomposition is a simulator-specific optimization process; subsequent users of a SIF data set will have their own simplification and decomposition rules, based upon their own image generator characteristics, and so they will need to perform their own optimization anyway. Also, in the process of decomposition, the accuracy of the original feature is lost, so even the reassembled polygons may bear little resemblance to the original feature. Since the decomposition of source features, then, both decreases the quality of the data set and increases the effort associated with using it, it will not be permitted in SIF data sets accepted by the SDBF.

50.1.3.2.5.2.3 Radar Characteristics. The reflectivity values contained in a specific radar data base are often optimized to create a realistic simulation of one particular radar set. For more general application, it is helpful to know the characteristics of the display for which it was intended.

50.1.3.2.5.3 Gridded Data Section. Self-explanatory.

50.1.3.2.5.3.1 Terrain Post Spacing. In that the SSDB stores terrain in an arc-second spacing scheme, it is necessary for incoming SIF data to comply with this format. If this terrain grid is interpolated from some other scheme (such as Universal Tranverse Mercator (UTM)), it will not necessarily retain the accuracy of the source from which it was generated; for this reason, the producer needs to notify the SDBF when the data has been resampled, so that a determination can be made whether it meets the accuracy standards of the SSDB.

50.2 SIF/DP Data Base Format. The purpose of this section is to define the overall SIF/DP data base format, in both a logical form as well as a physical tape form. The SIF/DP Data Base Format was designed to be nearly identical to the SSDB. It was meant to be processed by a copy of the SDBF system or an equivalent system. Thus, the data provided through SIF/DP consists of a dump of the SSDB with some additional control information.

50.2.1 SIF/DP Data Base Structure. SIF/DP consists of four general classes of simulator data. These are terrain, culture, models, and texture. For each application of the SIF/DP standard, these classes of data shall be included or excluded as appropriate to the sending and receiving applications.

50.2.1.1 Logical Format. (Self-explanatory.)

50.2.1.1.1 Data Base. (Self-explanatory.)

50.2.1.1.2 Section. (Self-explanatory.)

50.2.1.1.3 File/Record/Field/Item. Usually, a field consists of a single item. An example of a field with more than one item is a vertex field where each of the coordinates (X, Y, Z) defining the vertex are items.

50.2.1.2 Physical Format. (Self-explanatory.)

50.2.1.2.1 Data Order. The first save set on the first tape of the data base consists of a single file: the mandatory SIF/DP Data Base Header File. It contains control information, including counts of various data entities as well as the file name of each file in the data base. Each of the following sections consists of one or more save sets, each of which consist of one or more files. Each of the four sections is optional, and their existence is indicated within the SIF/DP Data Base Header File. For further details on the content of the files within the four main data sections, one should consult the SSDB Data Base Design Document (DBDD).

50.2.1.2.2 Physical Tape Format. It is a subset of the ANSI standard. According to this standard, volumes are written and read on 9-track magnetic tape drives only. The standard specifies the format, content, and sequence of volume labels and file labels. All labels must consist of ASCII characters. Four file/volume configurations are supported. They are single file/single volume; single file/multi-volume; multi-file/single volume; and multi-file/multi-volume. A SIF/DP data base may span several tape volumes. The ANSI standard specifies a maximum block size of 2048 bytes; however, SIF/DP allows larger block sizes. Larger block sizes tend to be more optimal in tape usage. It is recommended that SIF/DP data base creators use as large a block size as possible, given the processing capabilities of the systems exchanging data bases. The SDBF system supports up to the maximum block size. Specific save set names used by convention are listed in the format descriptions. Those specific save set names listed in each of the sections are required. For more details, one should consult the specified ANSI standard and/or VAX/VMS documentation, including the VAX VMS Magnetic Tape User's Guide.

50.2.2 SIF/DP File Formats. Some files are ASCII while others are VAX/VMS binary indexed files or VAX/VMS binary sequential files. The binary files use VAX/VMS format for integer and floating point values. The files in SIF/DP are either identical or nearly identical to their counterparts in the SSDB. Some files may have additional information added to them for SIF/DP purposes.

50.2.2.1 Header File. This section defines the detailed file, record, and field structure of the SIF/DP data base header file format. This format is used to provide general information on the contents of a SIF/DP database.

50.2.2.1.1 Header Data Encoding. This file contains general transmittal, identification, and directory information. The intent of the file is to allow a SIF/DP user to plan for the data to be input from the data base media. Information, including areas of coverage and file names, are provided for models, culture, terrain, and texture. A compressed form of ASCII has been chosen for the data base header file by the SIF/DP designers. Since many records are optional and the number of records of a certain type may vary, a method is needed so that one knows the type of record being read. The keyword approach has been adopted in the SIF/DP Data Base Header File. To minimize the impact of additional storage, keywords have been limited to two ASCII characters.

50.2.2.1.2 Header Section Structure. (Self-explanatory.)

50.2.2.1.3 Header File Structure. This mandatory file shall contain general transmittal, identification, and directory information concerning the SIF/DP data base to follow. It shall be the first file on the first tape volume. The order of data in the SIF/DP Data Base Header File is as specified below. The order in which the file names appear in this file is the required order in which the files shall appear in the data base. All records in this file are defined in this section. Data field definitions are provided in the Data Dictionary appendix.

50.2.2.1.3.1 SIF File Identifier Record. This mandatory record contains identifiers indicating the type of file.

50.2.2.1.3.2 Transmittal Description Record. This mandatory record contains identification information for the entire data base.

50.2.2.1.3.3 Data Directory Record. This mandatory record contains directory information regarding the entire data base.

50.2.2.1.3.4 2D Static Model Library Header File Name Record. This record is mandatory only if 2D static models exist in the data base. The existence of 2D static models is indicated by a count in the Data Directory Record. If they do exist, there shall be exactly one of these files. The record contains the file name for a single 2D static model library header file.

50.2.2.1.3.5 2D Static Model Entry Record. The record contains identification and descriptive information for a single 2D static model.

50.2.2.1.3.6 3D Static Model Library Header File Name Record. This record is mandatory only if 3D static models exist in the data base. The existence of 3D static models is indicated by a count in the Data Directory Record. If they do exist, there shall be exactly one of these files. The record contains the file name for a single 3D static model library header file.

50.2.2.1.3.7 3D Static Model Entry Record. The record contains identification and descriptive information for a single 3D static model.

50.2.2.1.3.8 3D Dynamic Model Library Header File Name Record. This record is mandatory only if 3D dynamic models exist in the data base. The existence of 3D dynamic models is indicated by a count in the Data Directory Record. If they do exist, there shall be exactly one of these files. The record contains the file name for a single 3D dynamic model library header file.

50.2.2.1.3.9 3D Dynamic Model Entry Record. The number of these records shall correspond to the number of 3D Dynamic Models Field, found in the Data Directory Record. The record contains identification and descriptive information for a single 3D dynamic model.

50.2.2.1.3.10 Culture Cell Header Control Record. This record is mandatory only if culture exists in the data base. If culture is provided, then there shall be one of these records for each cell. The number of cells corresponds to the count given in the Data Directory Record. The record contains the file name for the culture cell header file for a specific cell. This record also contains the identifying southwest corner and the number of culture manuscripts within this cell.

50.2.2.1.3.11 Culture Manuscript Data File Names Record. This record is mandatory for each culture manuscript in the data base. The number of these records for a given cell is provided by the count found in the Culture Cell Header Control Record. The record contains the file name for each file with information for a single culture manuscript.

50.2.2.1.3.12 Terrain Cell Header Control Record. This record is mandatory only if terrain exists in the data base. If terrain is provided, then there shall be one of these records for each cell. The number of cells corresponds to the count given in the Data Directory Record. The record contains the file name for the terrain cell header file for a specific cell. This record also contains the identifying southwest corner and the number of terrain manuscripts within this cell.

50.2.2.1.3.13 Terrain Manuscript Data File Names Record. This record is mandatory for each terrain manuscript in the data base. The number of these records for a given cell is provided by the count found in the Terrain Cell Header Control Record. The record contains the file name for each file with information for a single terrain manuscript.

50.2.2.1.3.14 Generic Texture Entry Record. This record is mandatory for each generic texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single generic texture.

50.2.2.1.3.15 Stage 3 Specific Model Texture Entry Record. This record is mandatory for each stage 3 specific model texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single stage 3 specific model texture.

50.2.2.1.3.16 Stage 2 Specific Model Texture Entry Record. This record is mandatory for each stage 2 specific model texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single stage 2 specific model texture.

50.2.2.1.3.17 Stage 1 Specific Model Texture Entry Record. This record is mandatory for each stage 1 specific model texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single stage 1 specific model texture.

50.2.2.1.3.18 Stage 3 Specific Areal Texture Entry Record. This record is mandatory for each stage 3 specific areal texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single stage 3 specific areal texture.

50.2.2.1.3.19 Stage 2 Specific Areal Texture Entry Record. This record is mandatory for each stage 2 specific areal texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single stage 2 specific areal texture.

50.2.2.1.3.20 Stage 1 Specific Areal Texture Entry Record. This record is mandatory for each stage 1 specific areal texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single stage 1 specific areal texture.

50.2.2.1.3.21 SMC/FDC Texture Entry Record. This record is mandatory for each SMC/FDC texture in the data base. The number of these records is provided by the count found in the Data Directory Record. The record contains identifying and descriptive information for a single SMC/FDC areal texture.

50.2.2.2 Model Data. The purpose of this section is to define the detailed file, record, and field structure of the SIF/DP CSG and polygonal model data format.

50.2.2.2.1 Model Data Encoding. The CSG approach to modeling describes physical objects using boolean combinations of a few simple primitive solids, such as spheres and cylinders. The CSG approach was selected as being significantly more generic (i.e., system-independent) than the polygonal (i.e., surface-based) approach typically used by simulator vendors. Although CSG primitives are volumetric (3-dimensional), 2-D parts within a model may be represented by modeling the objects as one or more thin plates, which may then be "collapsed" into a flat plane by the CDBTP during polygonization. In those cases where a shape (either 2-D or 3-D) is too irregular to be efficiently represented using standard solids, the SDBF software design supports definition of free-form polygonal cross-sections on a vertex-by-vertex basis. These cross-sections may then be used to generate 3-D volumes via CSG operations such as 'sweep' or 'revolve'. The SDBF uses a commercial off-the-shelf (COTS) product called ICMGMS (Interactive Computer Modelling Geometric Modelling System) as the basic toolkit for modeling application software.

50.2.2.2.2 Model Section Structure. The SSDB supports storage of each model at up to nine levels of detail (LODs). The resolutions for the LODs may vary from model to model since the way a model is built depends on the specific application for which it was intended. In the SSDB, a commercial software product (ICMGMS) with a particular implementation of standard CSG commands has been used. The polygonal geometry of each model is represented as a set of surface polygons in 3-D space. The perimeter of each polygon is described by a set of coordinates or vertices. The polygon is implicitly closed. Vertices are listed in counter-clockwise order. Each surface or polygon may have descriptive and rendering attributes associated with it. A wide range of fields are available to describe attributes specific to radar, visual, and infrared simulation, as well as general attributes applicable to all three. Each polygon may reference texture maps from an associated Model Texture Library. The model library structure also supports composite models in which one model references another as a component.

50.2.2.2.3 Model File Structure. (Self-explanatory.)

50.2.2.3 Culture Data. The purpose of this section is to define the detailed file, record, and field structure of the SIF/DP culture data format.

50.2.2.3.1 Culture Data Encoding. The SIF/DP culture format is nearly identical with the logical formats of the Project 2851 Standard Simulator Data Base (SSDB). This is natural for a format intended to support distributed maintenance of the SSDB. The SSDB stores culture data in a vector graphics format (points, lines, and polygons). The format is comparable to DMA Digital Feature Analysis Data (DFAD), but considerably extended in terms of point precision and descriptive attributes supported. Another valuable feature of the SSDB format is its support for a FACS-like attribute coding scheme. Modeled after DMA's Feature Attribute Coding System, the FACS is a system of self-defining feature categories and attributes, which gives the SDBF flexibility to add new feature categories and attributes to SIF/HDI as the need arises.

50.2.2.3.2 Culture Section Structure. The general data architecture for SSDB culture files is modeled after standards and conventions established by the Defense Mapping Agency. The inclusion of culture within a SIF/DP database may be toggled such that no culture is sent or all culture within the specified area of coverage is sent.

50.2.2.3.3 Culture File Structure. (Self-explanatory.)

50.2.2.4 Terrain Data. The purpose of this section is to define the detailed file, record, and field structure of the SIF/DP terrain data format.

50.2.2.4.1 Terrain Data Encoding. The SIF/DP terrain format is nearly identical with the logical formats of the Standard Simulator Data Base (SSDB). This is natural for a format intended to support distributed maintenance of the SSDB. The SSDB stores terrain data in a systematically spaced grid format comparable to Defense Mapping Agency (DMA) Digital Terrain Elevation Data (DTED), but supports a much wider range of post spacings, to support simulator applications with widely varying resolution requirements.

50.2.2.4.2 Terrain Section Structure. The inclusion of terrain within a SIF/DP database may be toggled such that no terrain is sent or all terrain within the specified area of coverage is sent.

50.2.2.4.3 Terrain File Structure. (Self-explanatory.)

50.2.2.5 Texture Data. The purpose of this section is to define the detailed file, record, and field structure of the SIF/DP texture format.

50.2.2.5.1 Texture Data Encoding. The SIF/DP texture format is intended to support distributed maintenance of the texture files within the Standard Simulator Data Base (SSDB). Therefore, the data format will be very similar, if not identical, to the internal binary format used in the SSDB.

50.2.2.5.2 Texture Section Structure. Each texture library has a toggle associated with it to determine the texture libraries to be sent. Textures to be sent are determined by their areas of coverage except for generic textures.

50.2.2.5.3 Texture File Structure. (Self-explanatory.)

50.2.3 <u>SIF/DP Data Base Content</u>. SIF/DP was designed for the express purpose of providing a highly efficient interface to the SSDB, as it is stored internally by the SDBF data base generation system. In doing so, it was decided that virtually no tailoring or filtration of the SSDB would be supported, since these operations would impede the rapid generation and utilization of SIF/DP data sets. Thus, by definition, a SIF/DP data set is a direct copy of the SSDB from which it was created. The only options available to SIF/DP producers and consumers are the "toggles", which determine whether or not to incorporate each of the major sections.

60 NOTES

60.1 <u>Notes on Appendix A</u>

60.1.1 <u>General Design Approach</u>. The formats of the different data fields defined within the SIF/HDI and SIF/DP files require the definition of different data structures within the data dictionary and methods for storing these data fields within a SIF/HDI tape file.

a. Data fields specified within the SIF/HDI formats and SIF/DP formats can be divided into two categories, atomic level data fields and composite level data fields. The atomic level data fields are defined as containing only one data value, and the composite level data fields are defined as containing two or more data values.

b. When storing an atomic level data field into a SIF/HDI ASCII tape file, the format is defined as being the value of the data field followed by the field separator mark (an ASCII Null mark '00'). An example is shown below:

    Field_100Field_200Field_300...

c. When storing a composite level data field into a SIF/HDI ASCII tape file, the format is defined as being the value of the first data item followed by an intra-field separator (a blank character ' ') followed by the value of the next data value in the field. If there are more data values contained within the field, the second data value is followed by an intra-field separator followed by the next data value, and so on until all data values for the data field have been written to the tape file. After all data values have been written, the data field is terminated with an ASCII null character. Examples are shown below:

    Group_1_Field_1 Group_1_Field_2 Group_1_Field_300...

    Group_1_Field_1 Group_1_Field_200Group_2_Field_1 Group_2_Field_200...

d. When storing either an atomic level data field or a composite level data field into a SIF/HDI binary tape file, there are no field separators nor any intra-field separators.

e. Except for where specially noted, all data fields are in an ASCII string format. Thus, all fields marked as type BOOLEAN, ENUMerated, REAL6, REAL10, INTeger, etc., are actually converted to ASCII strings. If a data field is truly binary, then its type shall indicate this by the word "BINARY" in it (e.g., BINARY INT would indicate a binary integer type). Binary types are used in files that are entirely binary, while ASCII strings are used in files that are entirely in readable ASCII string format. Binary files tend to consist of long lists of coordinates.

f. For the data fields represented as ASCII strings, the lengths provided in this data dictionary indicate the maximum possible length (number of characters) for each data field that exists in the SIF Data Base Header File, the Model Section, and the Culture Section. For any particular instance of a data field in those areas, the actual length may be less due to the compression of blanks. (Note that blanks used as intra-field separators shall not be compressed.) For ASCII data fields in the Gridded Data Section for Terrain and Texture, the lengths shown in this data dictionary indicate the exact length of each field. The reason for this is that the Gridded Data Section follows the NITF standard which does not allow for blank compression within its header data. For all ASCII data fields, the length does not include the field separator; however, for composite level data fields, the length does include the intra-field separator(s).

g. For the data fields represented in a binary format, the lengths provided in this data dictionary indicate the exact number of bytes necessary to represent that type.

h. Since the Gridded Data Section treats field lengths differently than the other sections, it may be useful to distinguish these data fields from the rest. In order to extend the usefulness of this data dictionary, fields that occur only in the Gridded Data Section are denoted with "(GDS)" in the field name column following the name; fields that occur in the Gridded Data Section as well as others are similarly denoted with "(BOTH)".

i. There are several different types of data defined in this data dictionary. The integer type (INT) has several traditional ranges associated with it that correspond to the number of bytes often used to represent it (-128..127 for 1-byte integers, -32768..32767 for 2-byte integers, -2147483648..2147483647 for 4-byte integers). Some SIF integer types have other ranges due to the requirements of those data fields. Multiple integer fields can be grouped to form composite level data fields such as INT2D, INT3D, and INT4D.

j. The traditional real, or floating point, type is represented in scientific notation for ASCII strings and in standard ANSI/IEEE binary floating point notation (ANSI/IEEE Std 754-1985) for binary data. The number of significant digits is indicated by the type name (e.g., six significant digits in REAL6, ten significant digits in REAL10). Multiple real fields can be grouped to form composite level data fields such as REAL2D6, REAL3D6, REAL2D10, and REAL3D10.

k. The string type is an ASCII string of given length. In general, it can contain any free-form set of printable ASCII characters, including the blank character. There are some cases when only certain characters are eligible. For example, date fields would consist of numeric characters only, and file name fields would consist of alphanumeric characters and a few other eligible characters (as described previously in this document). The type name is STR.

l. The enumerated type is essentially the same as a string type except that there exists a small finite number of possible valid strings. These are shown in the range column of this data dictionary. The enumerated type consists of only alphanumeric characters and the underscore ("_") character. The length indicated for an enumerated type field indicates the length of the longest enumerated value for that type. Following the NITF conventions, enumerated type fields shall be padded with blanks when necessary in the Gridded Data Section; enumerated type fields in all other places in the SIF data base shall be of the exact length of that enumerated type value. The type name is ENUM.

m. The boolean type is a special enumerated type where the possible values are TRUE and FALSE only. The type name is BOOLEAN.

n. A few special types exist. For example, the HEX type is a 32-character string of hexidecimal digits (0..9, A..F). Some types are a combination of other types in that they consist of a series of data items of the same or different types. These are all considered composite level data fields.

60.2 Notes on Appendix B

60.2.1 FACS Commonality. The feature types and attributes described in the DMA glossary cover a majority of the different feature types and attributes required within the SIF. However, there is a need to expand on the DMA glossary for some of the more simulator specific attributes supported in the SIF format. The following paragraphs identify the FACS codes and P2851 specific FDC values that are to be used for the optional attributes for culture features or components of models as defined in the applicable sections of the document.

60.2.2 FACS Codes. This section provides a mapping from the different SIF optional attribute fields to the actively supported FACS codes. Originator codes are assigned to SIF users as described in the main body of this document, section 4.2.1, Physical Tape Labeling. Based on values identified in the User-Defined FACS tables for either culture or model data, and input from SIF users, the list of actively supported FACS codes can grow to include whatever descriptors are deemed necessary to attribute the SIF data. If a user-defined FACS attribute becomes part of the standard SIF FACS Code list, its FACS Code value may be modified and documented in a future revision to this specification.

60.2.3 SIF-Specific FDCs. Feature Descriptor Codes are primarily based on the FCode value that is described within the DMA FACS Glossary. As with the SIF supported FACS codes, the list of feature codes within the DMA glossary needs to be expanded upon to describe simulator specific features. The following list describes the FDCs that are specific to the SIF. For a complete listing of the FDCs that are to be used for describing features or models within the SIF, this list is to be combined with the FCodes in the DMA FACS Glossary.

a. The FDC list has the possibility for expansion in the future based on inputs from SIF users. Based on values identified in the FID/FDC Cross Reference tables for either culture or model data, and input from SIF users, the list of actively supported FDC codes can grow to include whatever descriptors are deemed necessary to describe the SIF data.